LOW-ORDER MODELS FOR DYNAMIC SYSTEMS

Julio Cantalapiedra

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

LOW-ORDER MODELS FOR DYNAMIC SYSTEMS

by

Julio Cantalapiedra

Thesis Advisor: G. J. Thaler

June 1972

Approved for public release; distribution unlimited.

Low-Order Models for Dynamic Systems

by

Julio Cantalapiedra
Lieutenant, Spanish Navy
B.S., Naval Postgraduate School, 1971

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 1972

ABSTRACT

An investigation directed at finding the best low-order model which approximates a given high-order system is presented. New insight is gained into the cost paid for the simplicity of the model and in the accuracy of the transient response of the model related to the magnitude of a cost function.

The problem is solved in the time domain by finding the best pole and zero locations of the model which minimize a defined error criterion. The computer is used to estimate these parameters, via a parameter minimizating program. A number of examples are included.

TABLE OF CONTENTS

ACKNOWLEDGEMENT

The author wishes to thank Professor George J. Thaler of the Electrical Engineering Department, Naval Postgraduate School of Monterey, California, for his valuable advice and guidance.

I. INTRODUCTION

The process of modeling a given system by a system of lower dynamic order can be applied in order to simplify systems prior to analysis or to simplify designs. The initial design of a complex system is more easily accomplished if we can determinate a low-order model which approximates a high-order system and provides useful information about the behavior and time-domain response of the system.

Most works on control and simulation theory are performed on the basis that a mathematical model of the plant to be simulated or controlled is known, although this model often can be more complicated than is really necessary. Simplification of the model seems to be adequate and useful in such complex dynamic systems. In the stated problem the two major cases are:

given 1. The measured time-domain transient response of a system to a known input driving function,

or 2. The input-output relation --exact transfer function-- or the describing set of differential equations of a linear, constant system of order "n"

find a linear, constant system of order "m" which best approximates the given system.

A number of mathematical methods and numerical techniques have been developed and proposed as an approach to the solution of the problem in specified and restricted

cases. All of them present radically different approaches but they can be divided into two main groups: one of them is to retain specified eigenvalues of the original system and to neglect those which do not contribute much to system response; the other is based on the estimation of a set of parameters of differential equations of specified order, the response of which approximates that of the original system when both have the same driving function as inputs.

The approximation problem considered by Meier [6] uses a quadratic performance index, the mean-square difference between the output of the given system and the output of the reduced model when a random process is the driving function to both systems. Necessary conditions for optimum parameter values are calculated by differentiating the performance index with respect to the parameters. The approximated model is obtained for a specified or chosen suitable value of the order "m".

The Meier and Luenberger's approximation [5] deals with the modeling by a system of fixed lower order and leads to a set of nonlinear algebraic equations which constitute the set of necessary conditions for the optimal approximating transfer function which minimizes an integral criterion.

Both methods require that the exact transfer function of the system must be specified and assume a stationary random process as input to exact and approximated systems.

Kuppurajulu and Elangovan [4] propose a method for reducing a high-order system to a number of simplified models

by dividing the total time of response into a number of smaller intervals. The reduction is based on retaining only those eigenvalues of the system which have dominant effect during the time interval of interest. Thus reduced models can be obtained for the initial, intermediate and final stages of transient response.

Davidson [8], [9] and Chidambara [9] also propose a method based on the retention of specified eigenvalues of the original system and they neglect those which are far from the jw-axis since they make little contribution to the total response, except at the beginning. It uses relationships from the time solution of the differential equations of the original model in order to develop the reduced model. It is a projection method: a projection operator is found when the projection error is minimized and then, a contraction process is operated for choosing a suitable basis of dimension "m" (m<n) and a suitable set of "m" variables.

Anderson's work is along the line of the second group, his method determines a reduced model by minimization of the mean-square error between the responses of both model and original system over a given interval. It uses the orthogonal projection theorem to minimize the sum of the square errors at the sampling instants, and the steady-state error between both responses is not forced to be zero.

There are other transfer function methods, as the one of Chen and Shich which is based on the continued fraction

expansion in polynomial form of the transfer function, truncating it after a suitable number of terms. It takes into account the fact that the quotients in the expansion --by the final value theorem-- are in order of decreasing contributions to the response as the steady state is approached.

But in general, all these methods require an exact knowledgment of the system transfer function and this requires that in practice we have to identify exactly the system in order to know its transfer function or the set of differential equations which define it. Unfortunately this is seldom achieved. It seems more realistic to use directly the measured input-output data for the system and determine from it the reduced model for the system.

Sinha and Pillie [1] present a method with this advantage based on the use of the matrix pseudoinverse to estimate the parameters of the model which minimize the sum of the squares of the errors between the responses of the system and the model to a given input. It only requires the measurements of the input-output data. Sinha and Beraznai [3] develop a more general method for any specified error criteria of optimization, providing flexibility by the choice of the criterion and using pattern search for the optimum approximation. This method determines the optimum low-order model for a given order of the model (a model of order "m" is known, $m<n$).

Fellows, Sinha and Wismatch [2] propose a simple method for reducing a high-order system to a second-order model. It selects the second-order model, that meets specified performance features of the step-response and which is a good approximation to the step-response of the actual system, based on the minimization of the integral of the square error criterion.

As can be seen during recent years methods and techniques have been developed and applied to replace a complex system by a simpler system, some of them choosing the modes to be discarded in the construction of the new low-order system by mathematical techniques and the other ones selecting the order "m" ($m<n$) of the model and calculating the optimum m^{th} order system which best fits the original by different methods and criteria for the minimization process.

In this thesis it is intended the search for the best model which fits the original by optimization in the time domain, or, if not, to answer the question of what will be the price which it is necessary to pay for the simplicity of the model if a previously selected order of the model is specified or desired.

Carrying out an exhaustive search and determination of all possible low-order models for specific high-order systems, hopefully seems feasible to provide a basis for comparison of the results obtained if the complete set of resulting data is analyzed and studied in order to establish a criterion which can be used for the selection of the model in a particular problem.

Powerful mathematical tools of optimization theory and suitable criterion for an optimum fit in the solution are used in the practical search of low-order model systems. Two general cases are investigated:

1. Modeling based on using directly the measured input-output data for the system.

2. Modeling when the exact system transfer function is specified.

In Chapter II of the thesis, a general description and the philosophy of the technique for the solution is given, and it is applied in Chapter III which is devoted to the study of four general examples selected as a guide for initializing the investigation in this aspect of the modeling problem. This search can be thought of as another approach to the determination of low-order models and as an effort directed to extend the research done in this sense.

II. TECHNIQUE FOR SOLUTION

A. GENERAL

In dealing with the modeling of high-order systems by an optimum low-order model two main cases can be expected:

1. The measured input-output data for the system is known.

2. The exact system transfer function or vector differential equation are given.

Some assumptions about the systems have to be made for the sake of convenience, although the same assumptions underline most of classical control. It is assumed that all systems are: linear, constant-coefficient, asymptotically stable and single-input and single output. Under these assumptions

a. the linear system will be easier to study than the general case.

b. a second-order system is clearly simpler than a fourth-order system.

c. all poles of the system will lie in the left half plane.

Also the optimum model system is that which minimizes the performance index previously selected.

The dynamic behavior of a second-order system has been thoroughly studied and is well known, thus finding a pair of complex dominant roots --for the high-order system would be equivalent to approximating the system by a second-order model. It is known that in design problems of high-order

feedback systems by algebraic methods, the designer selects a pair of complex root locations and then tries to locate two of the roots of the system at these locations and the other ones (undesired roots) at remote points far from the jw-axis in the left-hand side of the s-plane. In this way the transient response of the system closely follows the selected second-order system's response. The "dominant" character is achieved if:

a. the coefficients in the time response associated with the desired root are much larger than those associated with undesired roots.

b. the time constants of the undesired roots are much smaller than those of the selected pair.

For physical systems, closed-loop poles are seldom found at the origin and complex poles located far from the real axis produce oscillations in the time-domain response of a higher frequency than those located closer to the real axis.

In this work it is assumed that the step-response of the high-order system has an overshoot at least.

B. PHILOSOPHY OF APPROACH

A realistic approach to the derivation of the low-order model which better approximates the given high-order system is to perform it directly from the known response of the system to a specified driven function which may be obtained experimentally. If the system is to be operated always with a specific input signal, then this driving function can also be used as input of the model for comparison of both

responses. A step function as driving function is very common in dynamic system test and identification procedures, on the other hand, step-function excitation often has the important advantage of simplicity.

Thus, the problem can be restated as the determination of the transfer function of the low-order model which is an optimum approximation to the step-response of the actual high-order system, based on some criterion of goodness.

It was felt that a convenient approach leading to the problem solution is the determination of the l^{th} optimum model ($l<n$) satisfying some specification of criterion rather than the determination of a specific low-order system, i.e. a second-order model.

1. Selecting the System Response

A typical time characteristic, acceleration curve $x_{out}(t)$ of an actually existing dynamic element is shown in Fig. II.1. In a general case an acceleration curve consists of a delay portion of duration t_d, a concave portion T_1, a constant-velocity portion T_2, a convex portion T_3 and the steady-state portion.

Figure II.2 shows typical unit-numerator step-response patterns for systems up to sixth order.

If the step-response of a system takes the form shown in Fig. II.3, it can often be approximated as a time delay plus the response of a second-order system with two complex roots.

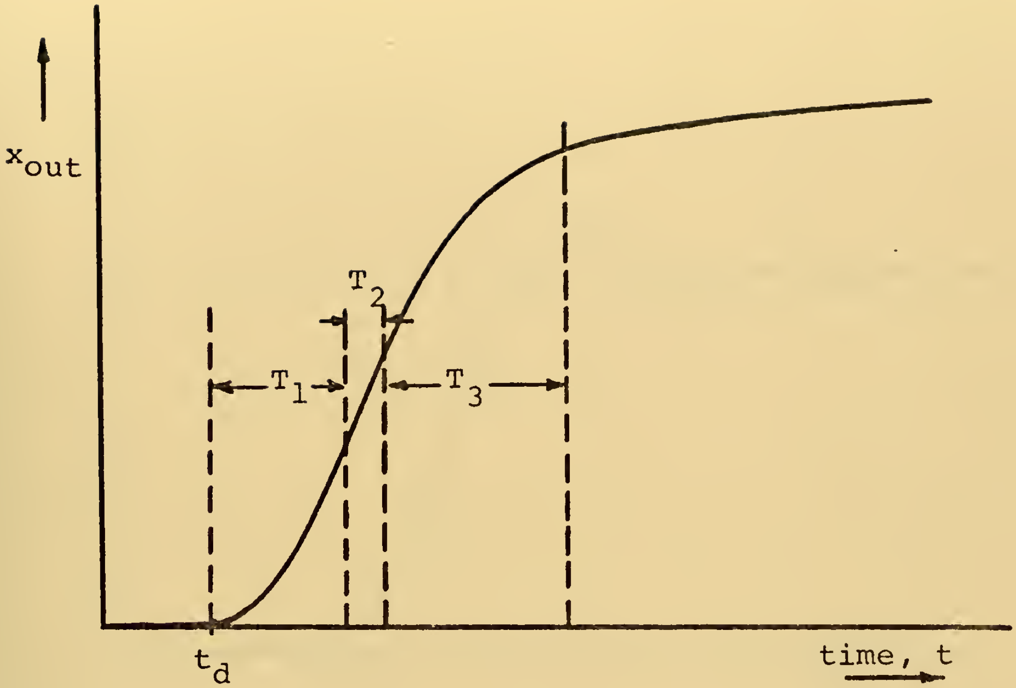

Figure II.1

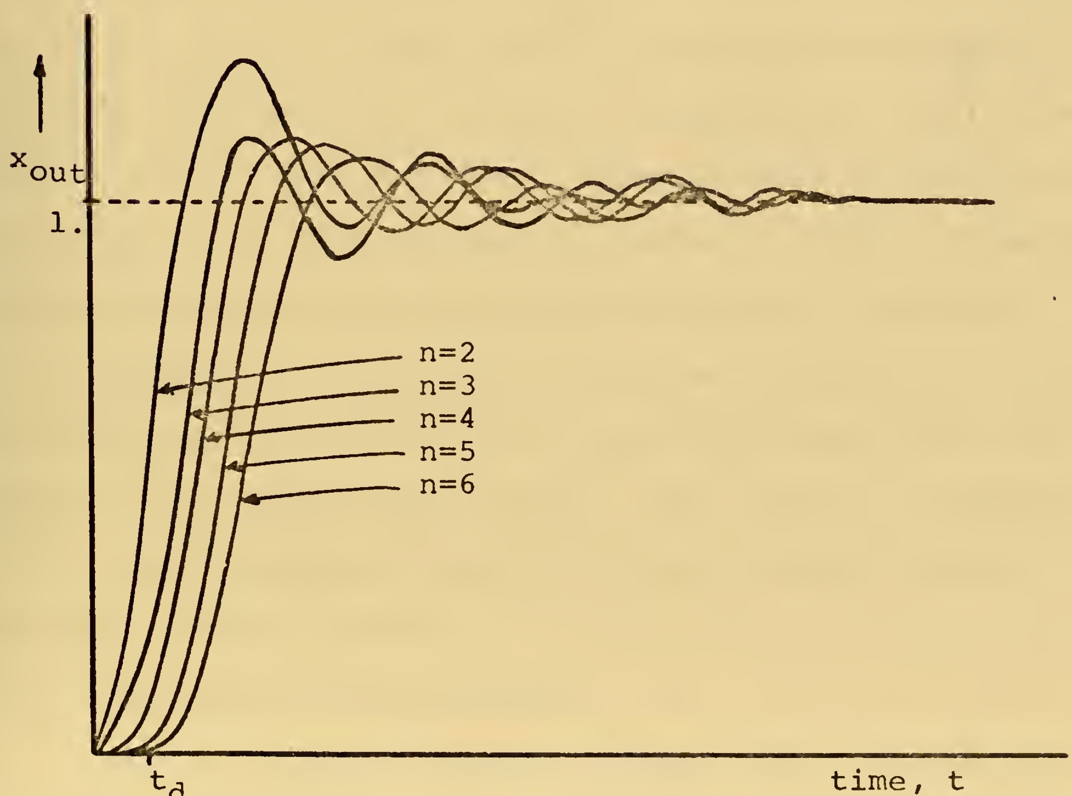

Figure II.2.

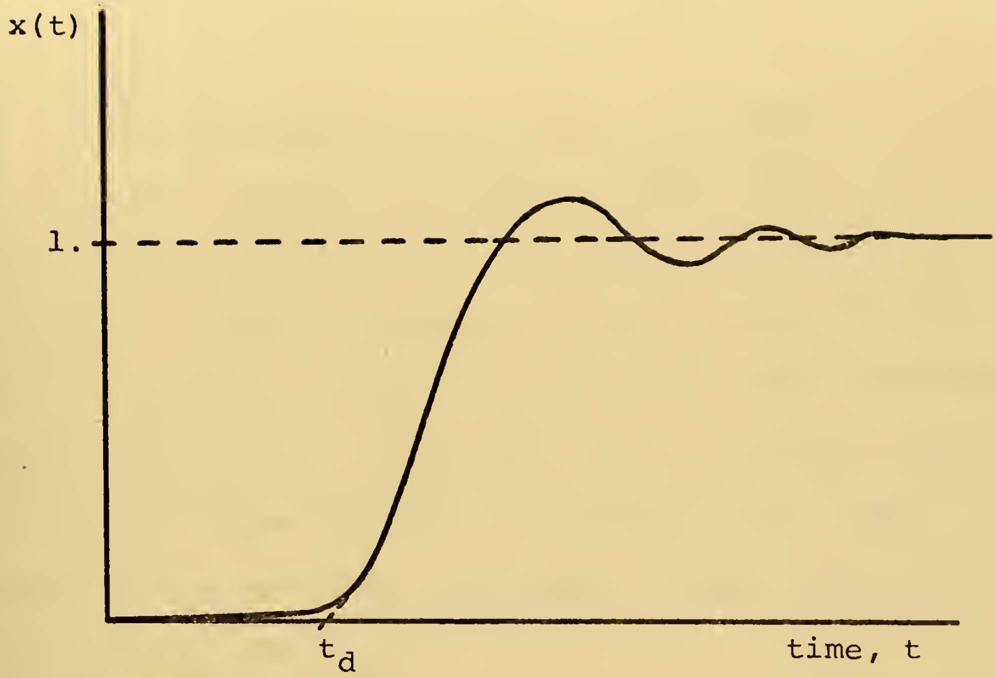

Figure III.3.

The common characteristic in the above step-responses is a time delay t_d. Very often in the study of practical high-order systems, this concept of introducing a pure time delay to account for the higher order permits interpretation and study of the step response in terms of relative stability and the nature of the corresponding frequency response.

Therefore, the high-order system response to a step input-function was selected as a pure time delay plus the response of a second-order system. The results in EXAMPLES I and II were obtained working with this assumed typical high-order system response.

2. Statement of the Problem

Let the given n^{th}-order linear, time invariant system be described by

$$\dot{\underline{x}}(t) = \underline{A}\,\underline{x}(t) + \underline{B}\,\underline{r}(t)\ ; \quad \underline{x}(0) = 0$$

$$\underline{y}(t) = \underline{C}\,\underline{x}(t) \qquad \text{(II.1)}$$

For the single-input case considered

$$\dot{\underline{x}}(t) = \underline{A}\,\underline{x}(t) + \underline{B}\,r(t)\ ; \quad \underline{x}(0) = 0$$

$$y(t) = \underline{C}\,\underline{x}(t) \qquad \text{(II.2)}$$

where $\underline{x}(t)$ is the n-dimensional state vector and the "dot" over a variable is used to indicate the time derivative (d/dt). $\underline{A}$ is a "n x n" matrix, $\underline{B}$ is an n-vector and r(t) is the scalar time function representing the input to the system (step-function). $\underline{C}$ is the n-vector and y(t) is the output of the system.

Consider a discrete set of values of y(t) taken over a suitable interval of time

$$Y = \{y_0,\ y_1,\ y_2, \ldots\ldots\ldots y_i, \ldots\ldots y_{ts}\} \qquad \text{(II.3)}$$

$$y_i = y(t_i)$$

which represents samples of the response of the system described by Eq. (II.2) to a step-function input. It may have been obtained by a measuring instrument connected to the actual system output with a sampling interval sufficiently small in order that no information be lost.

It is desired to find the best low-order model described by

$$\dot{\underline{x}} = \underline{A}_r\,\underline{x}_r + \underline{B}_r\,r(t)\ ; \quad \underline{x}_r(0) = 0 \qquad \text{(II.4)}$$

$$y_r = \underline{C}_r\,\underline{x}_r$$

whose set of output samples Y_r satisfies the condition of minimizing an error function $J(\underline{\alpha})$ given by

$$J(\underline{\alpha}) = f(y_i - y_{ri}) \tag{II.5}$$

where $\underline{\alpha}$ represents the N_c-vector of the free model parameters.

3. Error Function Criteria

The mean-square criterion was chosen as the function of the errors $(y_i - y_{ri})$. This performance index is suitable for many minimization problems and takes the form

$$J(\alpha) = \sum_{o}^{n_f} w_i \, (y_i - y_{ri})^2 \tag{II.6}$$

where n_f is a positive integer defined as

$$n_f = \frac{\text{final time of the calculations}}{\text{sampling period}} = \frac{t_f}{T} \tag{II.7}$$

and w_i is a weighting sequence which for the single-output case --assumed in this work-- is unity.

When (J) is minimized with respect to the free low-order model parameters in the bounded region defined by the constraints, the optimum values of the free parameters are found. Since one free parameter exactly locates one root of the model, when the error criteria is minimized the set of optimum parameters locate all the roots and the transfer function of the optimum model is determined.

The error criterion in the form

$$J(\underline{\alpha}) = \int_0^{t_f} (y_i - y_{ri})^2 dt \tag{II.8}$$

depends on t_f, but once t_f is chosen long enough and fixed, it does not affect the optimum parameter values obtained at the end of the minimization process. t_f was chosen as the settling time of the time-domain response of the high-order system.

C. METHOD OF SEARCH FOR BEST MODEL

1. Philosphy

The problem of approximating the high-order system by a low-order model in an optimum manner was solved for the specified error criterion (see Chapter (II.B.3)).

For the general l^{th}-model (Eq. II.4) the closed loop transfer function can be written as

$$G_r(s) = \frac{Y_r(s)}{R(s)} = \frac{b_m s^m + b_{m-1} s^{m-1} + \ldots\ldots\ldots + b_1\, s + b_0}{s^\ell + a_{\ell-1} s^{\ell-1} + \ldots\ldots\ldots\ldots + a_1\, s + a_0} \qquad \text{(II.9)}$$

or

$$G_r(s) = \frac{K(s+z_1)(s+z_2)\ \ldots\ldots\ldots\ldots(s+z_m)}{(s+p_1)(s+p_2)\ \ldots\ldots\ldots\ldots(s+p_\ell)} \qquad \text{(II.10)}$$

which is a function of the transfer gain and pole-zero configuration, which are the free parameters of the model.

Assume a complex system presenting a step-response which may have an overshoot. Then Eq. (II.10) may be modified

$$G_r(s) = \frac{K(s+z_1)(s+z_2)\ldots\ldots\ldots\ldots(s+z_m)}{(s^2+2\alpha_1\alpha_2 s+\alpha_2^2)(s+p_3)\ldots\ldots(s+p_\ell)} \qquad \text{(II.11)}$$

where

$$p_1+p_2 = 2\alpha_1\alpha_2$$

$$p_1 p_2 = \alpha_2^2$$

and $p_i>0$ (no pole at origin)

This low-order model is characterized and defined by a set of free independent parameters which can be chosen based on a specified error function criterion in order to optimize it. For given values of the pair "l" and "m"

($m>l>n$) the optimum l^{th}-order model which minimizes the error criteria is realized by an optimum choice of the free parameters. It is reasonable to expect the value of the error criteria between the responses of the actual system and the model, for each one of them, will decrease as the order of the model is increased. If a 3-dimensional parameter space is considered with coordinates axis

XX' : number of zeros of the model

YY' : number of poles of the model

ZZ' : values of the error criteria for optimum model

for a specific value of the transfer gain, the set of error criteria values for the multiples choices of both number of zeros and poles will be represented by a surface. Hopefully a minimum to this surface, if it exists, would represent the minimum of all minimum error values, in other words, it would define the "best" of all optimum low-order models for a specific value of the transfer gain.

The decision in the choice of the lower order approximated model which better represents the given system, if there is not a net minimum, can be done taking into account available criterion such as: limitations on the maximum order of the model because the limitations on the available model components, or sufficiency of the resultant error criterion (J) which satisfies the inequality

$$J - J_d > 0$$

where J_d is a minimum boundary --decision criterion-- for the error criteria (J). In this way, having plotted the

error criteria versus a specified number of poles and zeros it will be easy to decide which model accomplishes the decision criterion or which one is the best among all the low-order models not satisfying it.

Starting with a model with poles only (no zeros) the optimum set of parameters can be determined for the second-order model satisfying the selected error criteria, the order is then increased by one and a third-order model is determined in the same way, and so on. Plotting the results will show the variations of the error value with respect to the order of optimum models. Continuing with a model with one zero, the same search is performed increasing the number of poles by one. Then select models with two, three,.. zeros and for all of them, again, the optimum models with increasing number of poles are determined. The family of curves obtained are the major decision tool in the determination of the optimum low-order models.

2. <u>Constraints</u>

In order to get the optimum low-order system's transfer function it is necessary to find the optimum value of the transfer function gain, K, and the best locations of zeros and poles for the optimum model. Thus the number of free independent parameters at least is equal to $m+\ell+1$, where

m : number of zeros of the model

l : number of poles of the model

K : transfer function gain

As was stated in paragraph II.B.4, a 3-dimension parameter space is considered for representing the variations of the error criteria as a function of both number of zeros and poles.

In setting up constraints on the minimization process the transfer function gain may be kept constant or it may be allowed to vary as one of the parameters. If it is variable then a 4-dimensional space is needed. This not only makes it hard to find the minimum of the surface, but the concept of a varying gain is not consistent with most physical problems --at least not for control systems. Therefore in this study the gain is kept constant.

In practical systems the Body gain, i.e., the transfer function gain when written in Bode form, usually is fixed by specifications. In many applications it corresponds to the DC gain or zero frequency gain. If the transfer function is left in Root Locus form, i.e. (s+z)/(s+p), the Root Locus gain is a number that varies with both the number of zeros and poles, and as such does not usually have an apparent relationship with physical system performance characteristics.

Therefore the Body gain is kept constant in the search for the minimum, and it is also constrained to be positive.

In order to ensure the stability of the system the pole and zero location is restricted to the Left Hand Plane only, assuming no poles at origin.

3. Minimization in Parameter Space

The error criterion or performance index is minimized with respect to the free system parameters in the bounded parameter space defined by upper and lower boundaries on the parameter values and by the constraining inequalities and/or equations. When this is done it yields the best zero and pole locations for the given lower order model.

Various search techniques may be considered for the minimization process of the specific error criteria. Gradient methods are quite efficient in locating a minimum, although the necessity of calculating partial derivatives of arbitrary error criteria with respect to all the free parameters seems to be a disadvantage. Direct search involves evaluating the effect of sequential parameter changes in an organized manner.

A subroutine for the minimization of an arbitrary function by the complex method of M. J. Box [14] was chosen as a suitable method for this work.

An important problem associated with every search technique is the selection of the starting parameter values, because of their considerable influence on the convergence of the process and on the probability of locating the optimum minimum. If the step-response of the high-order system has an overshoot it is felt that the determination of a second-order system with a pair of conjugate poles (dominant roots) will be a good approach for the selection of the starting parameter values. From the step-response

some features usually can be specified: the maximum overshoot, the steady-state response, the settling time, the time to reach the first maximum; therefore, a second-order system that meets those specification approximately can be determined using the analytical expressions or the universal curves for the step-response as a function of the natural frequency (w_n) and the damping ratio (ζ). These approximate parameter values are used as starting values for the minimization program in order to get the optimum set of parameters for the second-order model and successive higher order models.

For the case in which the step-response has no overshoot a simple first order system can be determined by

$$G(s) = \frac{K}{s+p}$$

$p = 1/\tau$, τ = time required to reach 0.632A.

$K = A.p$, A = steady-state response to unit step.

for the selection of the starting parameter values in the minimization subroutine.

This thesis considers only step-response with an overshoot (more general case for physical systems).

D. PROGRAMMING

The performance index (error criteria) is minimized with respect to the free system parameters.

A subroutine has been used which finds the minimum in the given bounded region.

At each of the minimization procedure, Eqs. (II.4) and (II.8) are integrated from $t=0$ to $t=t_f$ by using a fourth-order Runge-Kutta method.[1] The optimum parameter and minimum error values are taken back to the main program which calls another subroutine (SIMUL) for the digital simulation of both high-order and low-order systems in order to get a graph representation of the time-domain responses and compare them.

If at the end of a minimization step the calculated parameters, or at least one of them, are on the limiting boundaries then the limiting boundary(s) can be relaxed and the minimization should be repeated again to obtain optimum parameter values for the new boundaries.

The original high-order system may be specified either by its output response to a given input step-function available at discrete uniform intervals of time, or by the closed-loop transfer function. In the latter case the main program computes the transient response at discrete uniform interval, stores it in an array "look-up" and the computation continues as in the first case.

The parameter starting values for the model to be used in the minimization program have to be given by the user.

A computational flow chart is shown in Fig. II.4.

---

[1]The increment time for the integration process should be equal to the sampling or integration times for the given high-order system.

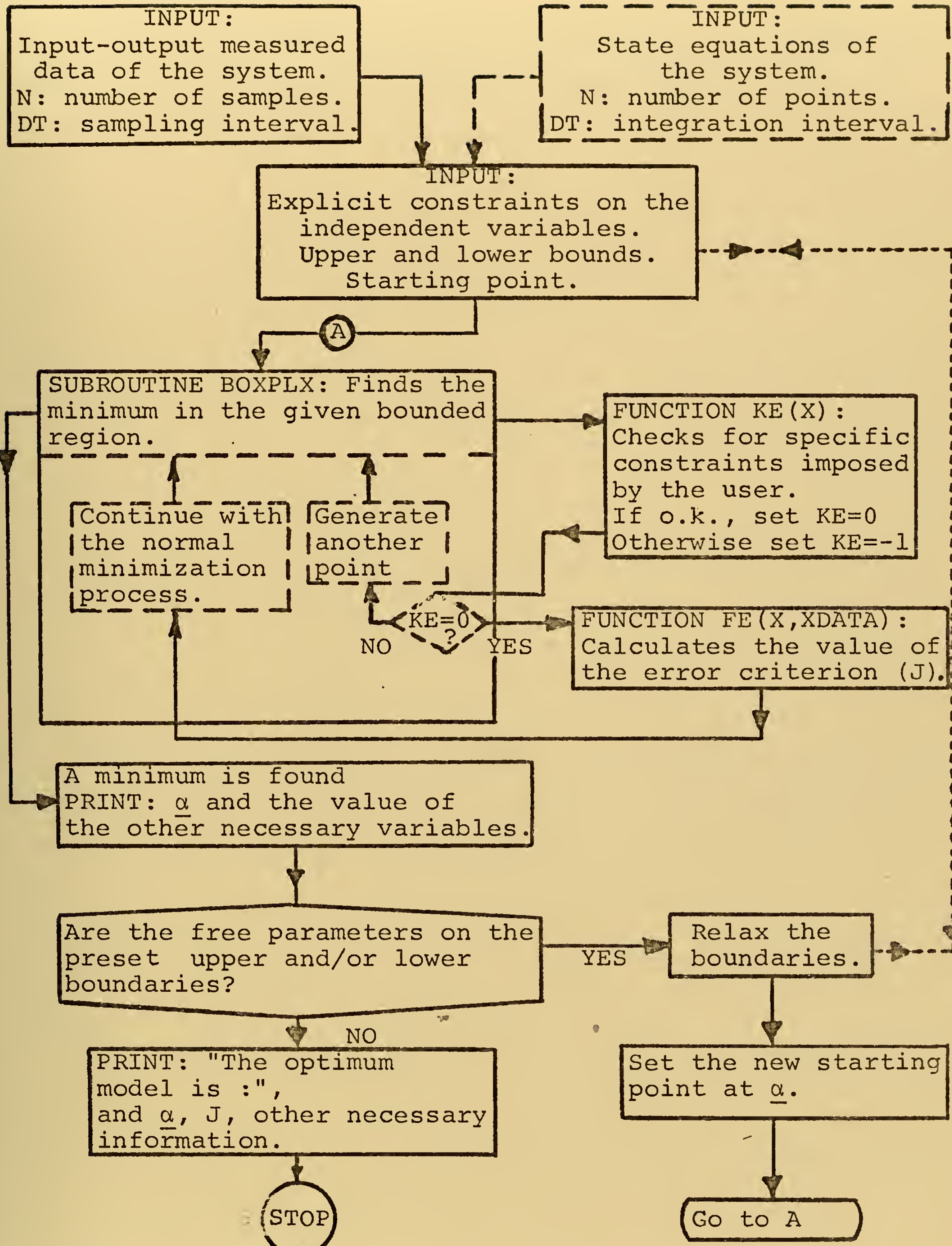

Figure II.4. Computational flow-chart.

Two computer programs which are written in Fortran IV, and used for two specific models of the Examples I and IV are given at the end of the thesis.

III. INVESTIGATION AND DERIVATION OF MODELS USING MEASURED INPUT-OUTPUT DATA FOR THE SYSTEM

A. GENERAL

It was felt that a convenient description of the actual high-order system measured input-output data has the form shown in Fig. II.3, namely a pure time delay plus the response of a second-order system. In this way the output of the system is obtained and sampled at suitable time interval (Δt).

The next step is interpreting the features of the step response more commonly specified, in order to select the starting parameters for the search of the model (as was indicated in Chapter II.3). Then the state equations of the possible model are integrated with respect to time with a step-input function, and the output of the model is obtained and sampled at Δt. These samples are compared with the values of the response of the system. An error criteria, which measures the deviation of the sampled model values from the sampled system values (from zero time to the selected settling time of the system) is defined. Minimization of this error criterion with respect to the model parameters yields the optimum model, for the given system description, using a specific order for the model.

The starting parameter values were taken as is stated in Chapter II, Section C.3.

B. EXAMPLE I

The input-output measured data for the testing system was taken as the step-response of a second-order system with

$$\zeta = 0.316$$

$$w_n = (10.)^{\frac{1}{2}} \qquad \text{(poles at: } -1. \pm j3\text{)} \qquad \text{(III.1)}$$

$$K = 10.$$

delayed an interval $t_d = 0.15$ seconds.

This response was computed by digital simulation from 0 to 5.2 seconds and 520 samples of this response (at intervals of 0.01 seconds) were used for the determination of optimum l^{th}-order models by minimization of the error criterion (Section II.B).

The investigation of possible models was carried out starting with l^{th}-order models ($l = 2, 3, \ldots 7$) with no zeros, and then repeating the search for models with one zero, then with two zeros, ending with models with three zeros. It is considered that 7 poles and 3 zeros are a good stop criterion for analyzing results.

As a constraint it is assumed that there is no error in steady-state response to a step input, keeping the Bode gain constant.

1. Two Poles No Zeros Model

a. Model response

The state equations of the second-order model can be written as

$$\dot{\underline{x}}_r(t) = \begin{bmatrix} 0 & 1 \\ -p_1 p_2 & -(p_1+p_2) \end{bmatrix} \cdot \underline{x}_r(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \cdot u(t) \quad ;$$

$$\underline{x}_r(0)=0 \tag{III.2}$$

where $u(t) = \mathbb{1}(t)$, $\underline{x}_r(t)$ is the state vector of the model and p_1, p_2 are the pole locations or free parameters of the model.

The output of the model is

$$\underline{y}_r(t) = [K \;\; 0] \cdot \underline{x}_r(t) \tag{III.3}$$

where K is the transfer gain.

Calling $\underline{\alpha}$ the N_c-dimensional vector representing the free model parameters (where N_c=number of poles + number of zeros), the state equations can be rewritten as

$$\dot{\underline{x}}_r(t) = \begin{bmatrix} 0 & 1 \\ -\alpha_2^2 & -2\alpha_1\alpha_2 \end{bmatrix} \cdot \begin{matrix} x_1 \\ x_2 \end{matrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \cdot u(t) \tag{III.4}$$

because they are chosen as parameters α_1 and α_2 -- the damping ratio and natural frequency.

b. Constraints

The only constraining equalities and inequalities are

$$K/\alpha_2^2 = \text{constant} = 1.$$

$$0<\alpha_1<0.5 \tag{III.5}$$

$$0<\alpha_2<5.5$$

as a boundary for the dynamic response.

c. Result of the minimization

The error criterion is written as

$$J(\underline{\alpha}) = x_J(t_f) = \int_0^{t_f} [y(t) - y_r(t)]^2 \cdot dt \tag{III.6}$$

where t_f = final time of the sampled complex system.

And its time derivative is

$$\dot{x}_J(t) = [y(t) - y_r(t)]^2 \tag{III.7}$$

The integration is performed with increment time Δt = sampling interval = 0.01 seconds.

The performance index $x_J(\underline{\alpha},t)$ is minimized by the computer subroutine with respect to the free parameters; the optimum parameter values and error criterion minimum value are tabulated in Table III.1, and the optimum time response of both the original and optimum second-order system (for a unit step input) are plotted in Fig. III.5.

2. Three Poles and no Zeros Model

a. In all this work it was assumed that the desired dynamic response of the model was represented by two complex roots and a certain number of real poles and zeros (because of the overshoot of the original system). Thus for this case the model transfer function

$$\frac{Y_r(s)}{U(s)} = G_r(s) = \frac{K}{(s+p_1)(s+p_2)(s+p_3)} \tag{III.8}$$

can be written as

$$G_r(s) = \frac{K}{(s^2+2\alpha_1\alpha_2 s+\alpha_2^2)(s+\alpha_3)} \tag{III.9}$$

where the free system parameters vector $\underline{\alpha}$ is represented by

$\underline{\alpha} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix}$ and each $\alpha_i = f(p_1,p_2,p_3)$.

The state equations for the model are written in the form

$$\dot{\underline{x}}_r(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_1 & -a_2 & -a_3 \end{bmatrix} \cdot \underline{x}_r(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u(t) \qquad \text{(III.10)}$$

and the output of the model is

$$\underline{y}_r(t) = \begin{bmatrix} K & 0 & 0 \end{bmatrix} \cdot \underline{x}_r(t) \ ; \quad \underline{x}_r(0)=0 \qquad \text{(III.11)}$$

where each $a_i = f(\alpha_1, \alpha_2, \alpha_3)$ and K is the transfer gain.

b. In setting constraints, again the Bode gain was kept constant and no error was permitted in steady-state response to a step input

$$\frac{K}{\alpha_2^2 \alpha_3} = 1. \qquad \text{(III.12)}$$

The third pole (α_3) was constrained to an upper bound of 100 and to be positive; flexibility was given in the computer program for changing this constraint if the upper boundary was reached. The minimization process showed that the boundary was not reached. The complex conjugate poles were constrained as in Section III.B.1.

Minimization of the objective function (J) yields as optimum parameter values the result tabulated in Table III.1 and plotted in Fig. III.1. Figure III.6 shows the time responses of both the system and the desired third-order model with no zeros.

3. <u>More Than Three Poles and No Zeros Models</u>

With the assumption of two complex conjugated roots for the model and following the same technique, search and determination of optimum models of order ranging from fourth

to seventh was carried out. The resulting optimum parameters and optimum error criteria are tabulated in Table III.1, and the minimum error criterion value is plotted versus number of poles in Fig. III.1.

4. Models With Zeros

The number of zeros of the model was increased by one and determination of the l^{th}-order model $(2<\ell<7)$ was investigated.

The state equations of the l^{th}-order system are written in the form

$$\dot{\underline{x}}_r(t) = \underline{A}_r \underline{x}_r(t) + \underline{B}_r u(t) \quad ; \quad \underline{x}_r(0) = 0 \qquad \text{(III.13)}$$

where

$$\underline{A}_r = \begin{bmatrix} 0 & 1 & \cdots\cdots & 0 \\ 0 & 0 & & 0 \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & 1 \\ -a_1 & -a_2 & \cdots\cdots & -a_\ell \end{bmatrix}$$

$$\underline{B}_r = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ 1 \end{bmatrix} \quad ; \quad u(t) = \mathbb{1}(t) \qquad \text{(III.14)}$$

and the output of the model is

$$\underline{y}_r(t) = \underline{C}_r \underline{x}_r(t) \qquad \text{(III.15)}$$

where

$$\underline{C}_r = K\,[b_1\ b_2\ \cdots\cdots b_m] \qquad \text{(III.16)}$$

In the same way as in section (2.a) the coefficients a_i and b_i are functions of the free parameters (poles and zeros) of the model system. They are expressed as the sum of combinations of products of the free parameters.

Each model was minimized with respect to the selected error criterion (Eq. (III.6)). The final time for the integration is taken as the final time sampled (t_f=5.2 seconds).

Again the starting parameter values were taken from the second order model and the boundaries from the same model with no zeros. Flexibility was given for changing the upper and lower bounds if any of them was reached and the result was a constrained minimum. Constant Bode gain and zero error between both steady-state responses were required.

The resulting free parameter values for each optimum model are tabulated in Tables III.2, III.3 and III.4 (and the corresponding error criterion values). A plot of the cost function (J) versus the number of poles, for constant number of zeros, is showed in Figs. III.2, III.3 and III.4.

Table III.5 is a comparison of the error criterion values for all models studied in this example and it shows the existance of a best-optimum low-order model. Figs. III.7 to III.18 show the time response of the system and several models.

5. Remarks

Looking at the tabulated results in Table III.5, which is a resume of the study done with this system and a clear basis of comparison between all the models, and at the plotted values of the performance index (J) versus number of poles, some facts can be clearly pointed out:

# Of Poles	2	3	4	5	6	7
ζ	.35	.288	.297	.303	.309	.31
w_N	2.734	3.145	3.148	3.148	3.138	3.137
p_3		6.24	7.8	9.35	13.46	14.25
p_4			37.1	29.7	25.4	24.98
p_5				87.24	41.57	69.
p_6					111.9	87.7
p_7						98.74
$Jx10^4$	131.	5.17	4.7	5.09	6.09	6.4

TABLE III.1. EXAMPLE I. Optimum models without zeros.

# Of Poles	2	3	4	5	6	7
ζ	.351	.25	.296	.301	.302	.303
w_N	2.72	2.99	3.15	3.147	3.141	3.144
p_3		2.69	8.46	8.2	8.64	8.2
p_4			12.84	9.22	10.34	15.57
p_5				22.36	37.4	38.72
p_6					87.1	61.66
p_7						91.75
z_1	199.8	5.	25.1	8.14	10.1	11.4
$Jx10^4$	140.	36.	4.4	4.8	5.	5.4

TABLE III.2. EXAMPLE I. Optimum models with one zero.

# of Poles	3	4	5	6	7
ζ	.274	.29	.294	.298	.296
w_N	3.126	3.147	3.149	3.145	3.149
p_3	4.96	8.46	7.595	9.67	7.4
p_4		9.52	14.327	10.086	15.44
p_5			20.428	48.778	35.34
p_6				53.34	65.55
p_7					105.6
z_1	34.76	28.13	18.11	14.68	14.55
z_2	148.9	33.57	26.7	49.55	36.13
$Jx10^4$	8.4	4.5	4.5	4.61	4.8

TABLE III.3. EXAMPLE I. Optimum models with two zeros.

# of Poles	4	5	6	7
ζ	.295	.281	.295	.297
w_N	3.152	3.139	3.151	3.003
p_3	9.7	6.67	9.372	0.014
p_4	10.87	8.6	10.104	20.78
p_5		153.9	43.54	25.76
p_6			120.91	47.52
p_7				82.91
z_1	49.8	20.19	19.	0.013
z_2	98.78	23.7	62.45	17.032
z_3	116.58	53.76	82.86	26.348
$Jx10^4$	4.24	4.402	4.406	4.47

TABLE III.4. EXAMPLE I. Optimum models with three zeros.

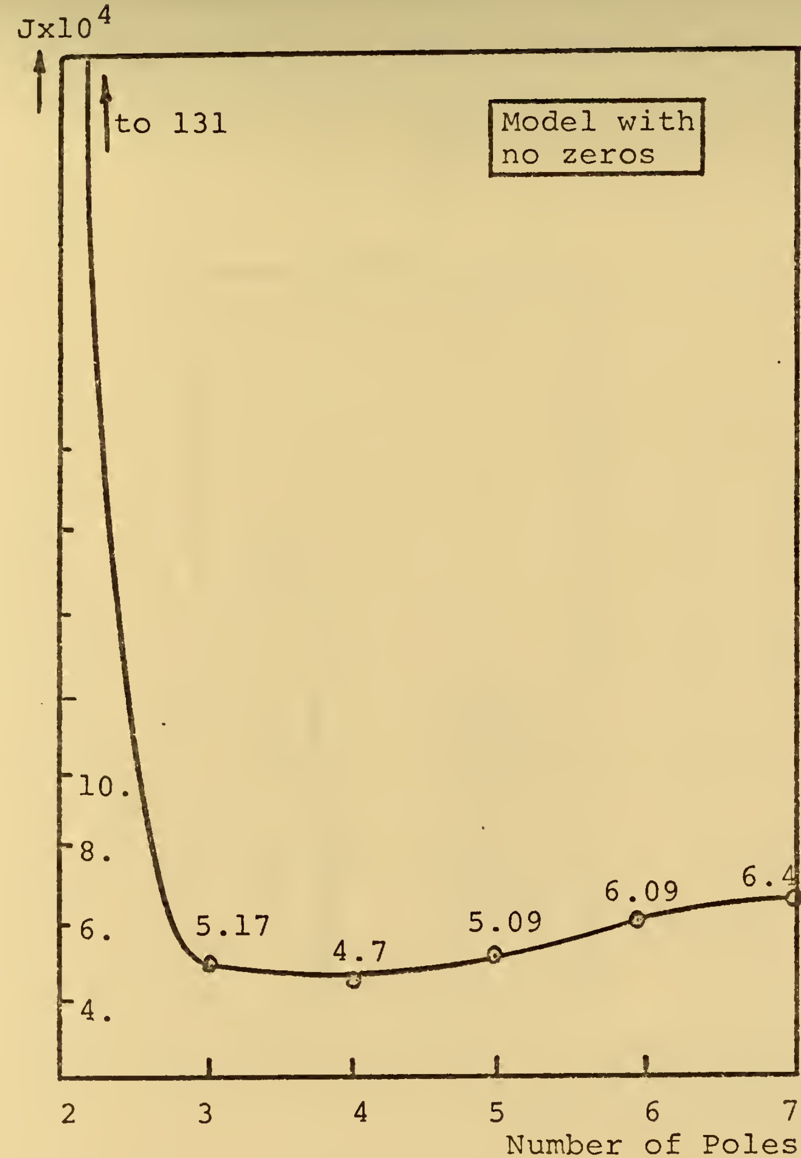

Figure III.1.

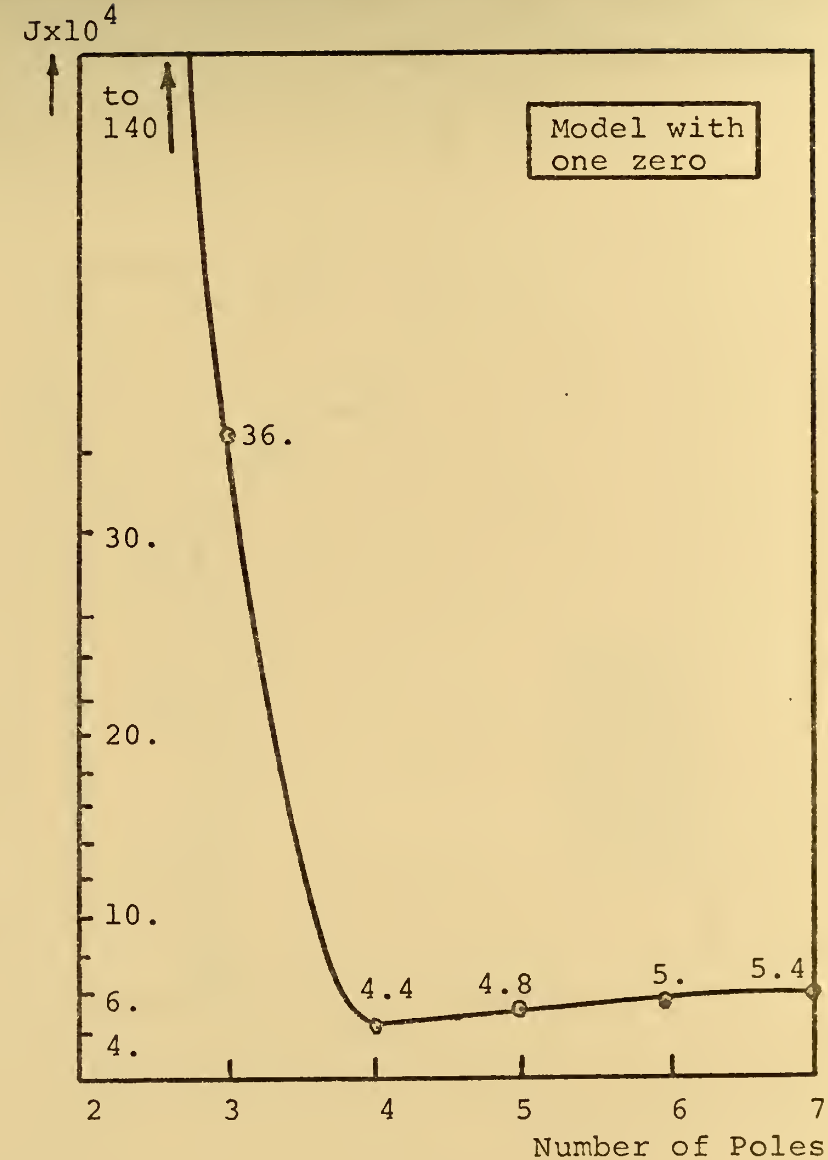

Figure III.2.

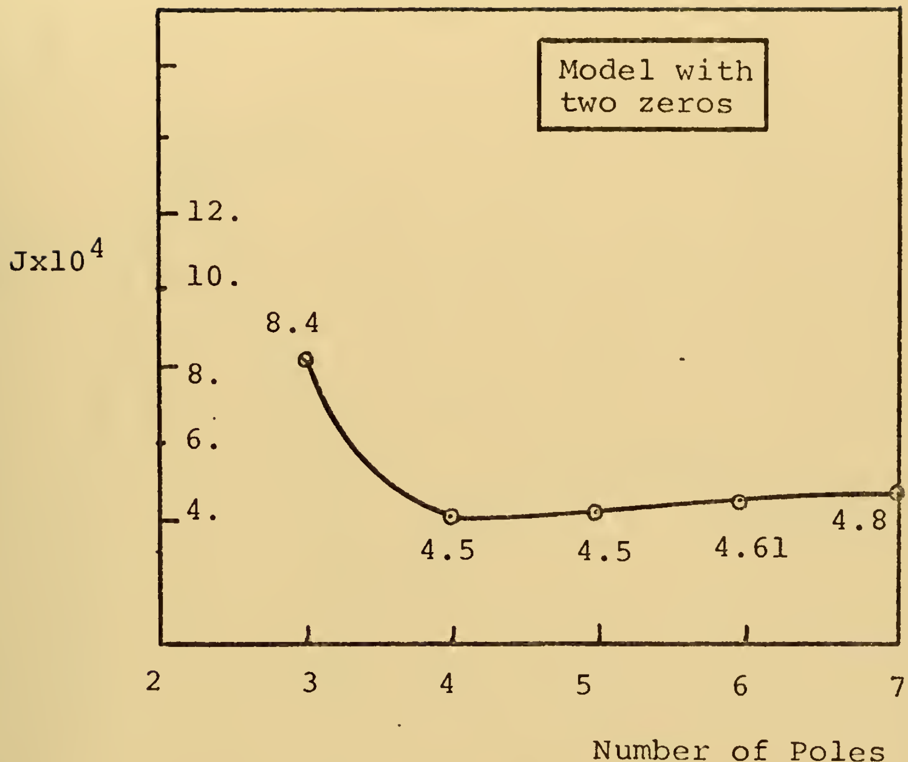

Figure III.3.

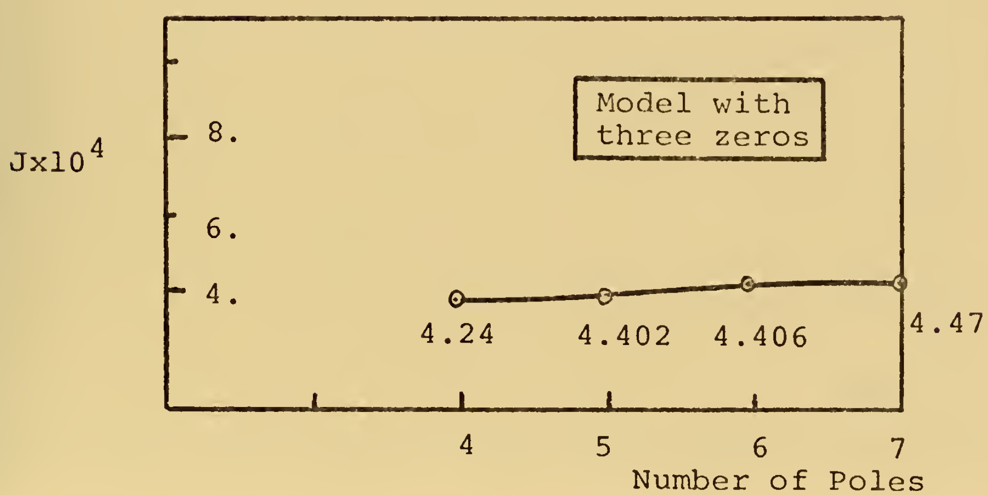

Figure III.4.

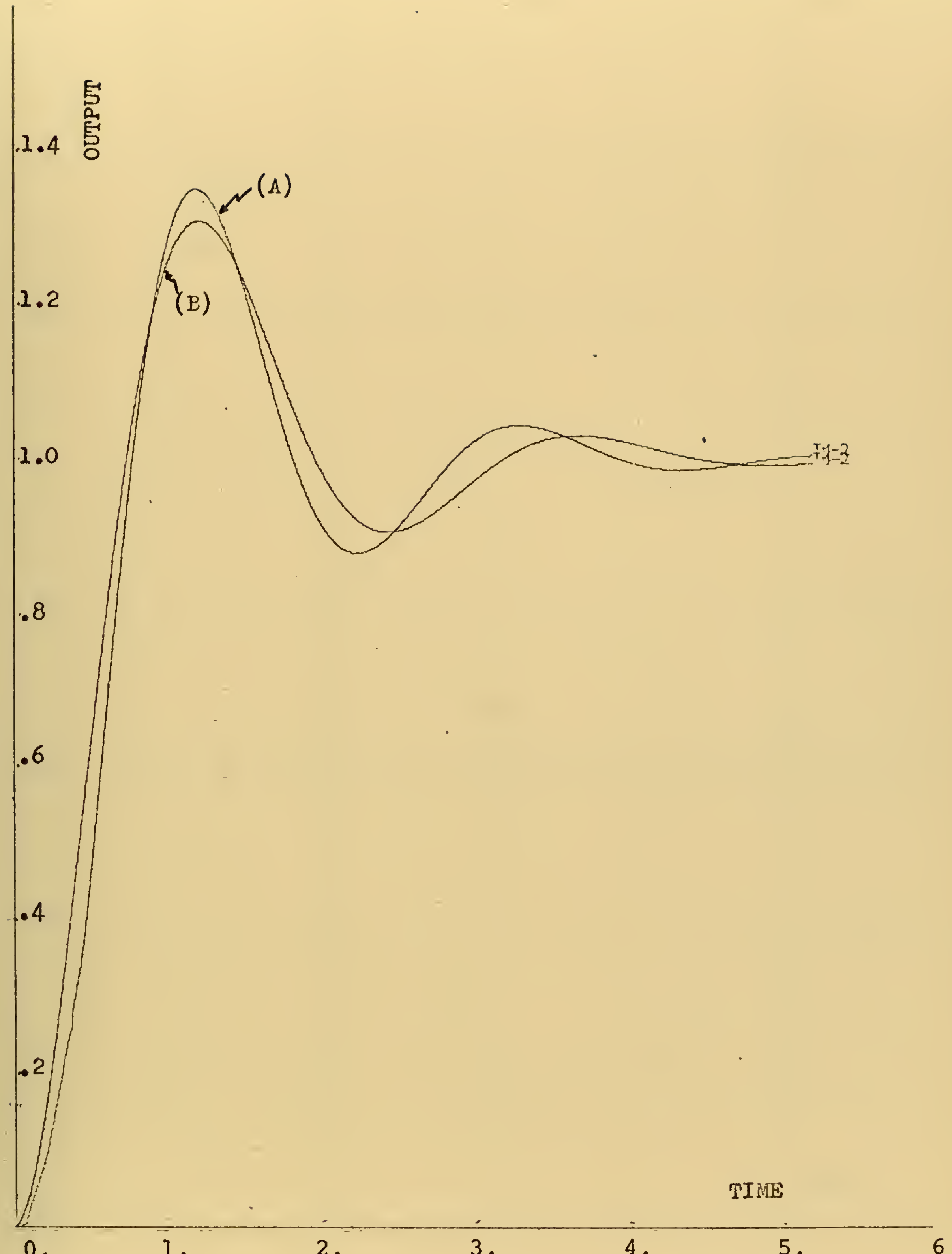

Figure III.5. EXAMPLE I. System's response (A) and the Two Poles no Zero model's response (B) to a unit step input.

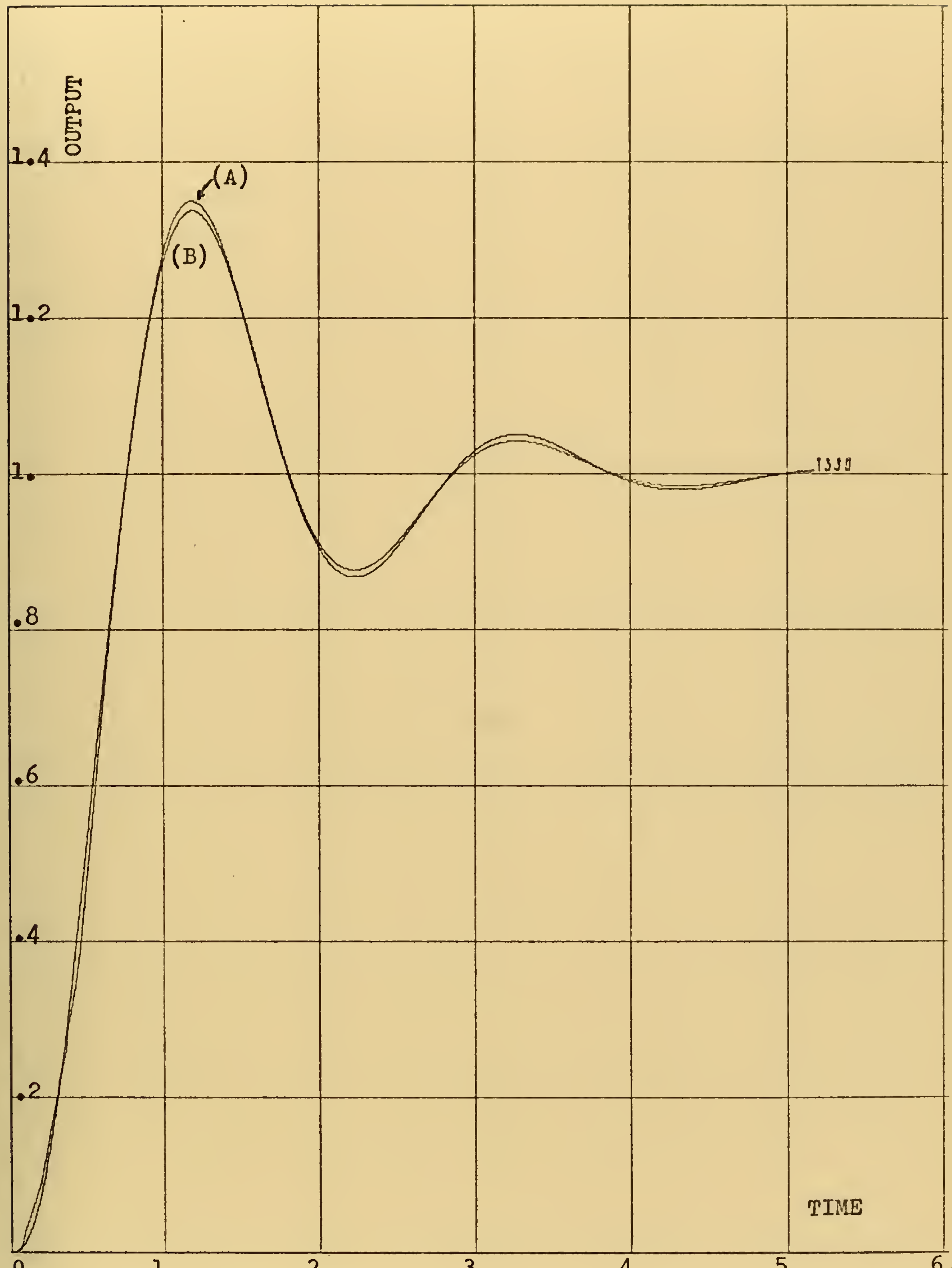

Figure III.6. EXAMPLE I. System's response (A) and the Three Poles no Zero model's response (B) to a unit step input.

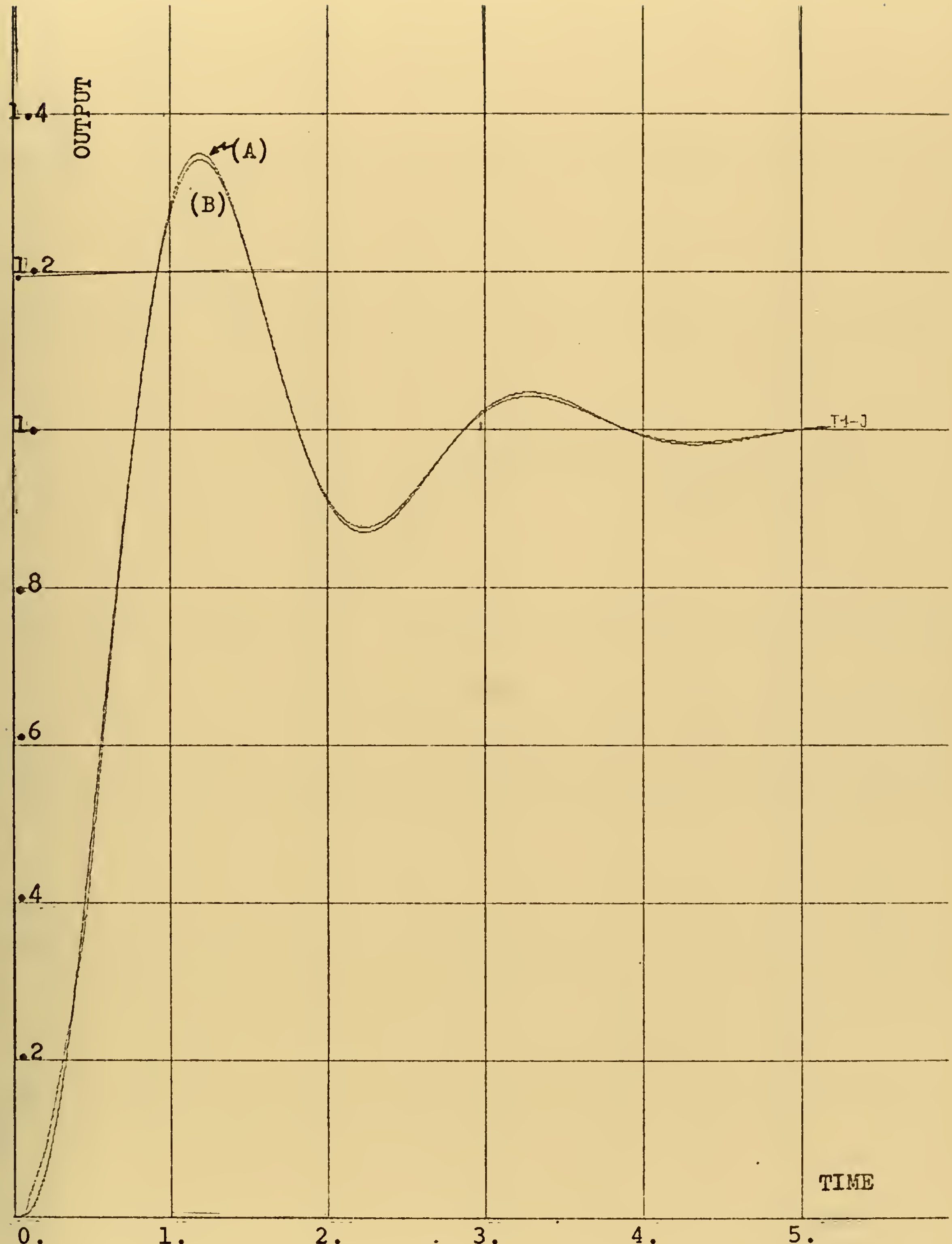

Figure III.7. EXAMPLE I. System's response (A) and the Four Poles no Zero model's response (B) to a unit step input.

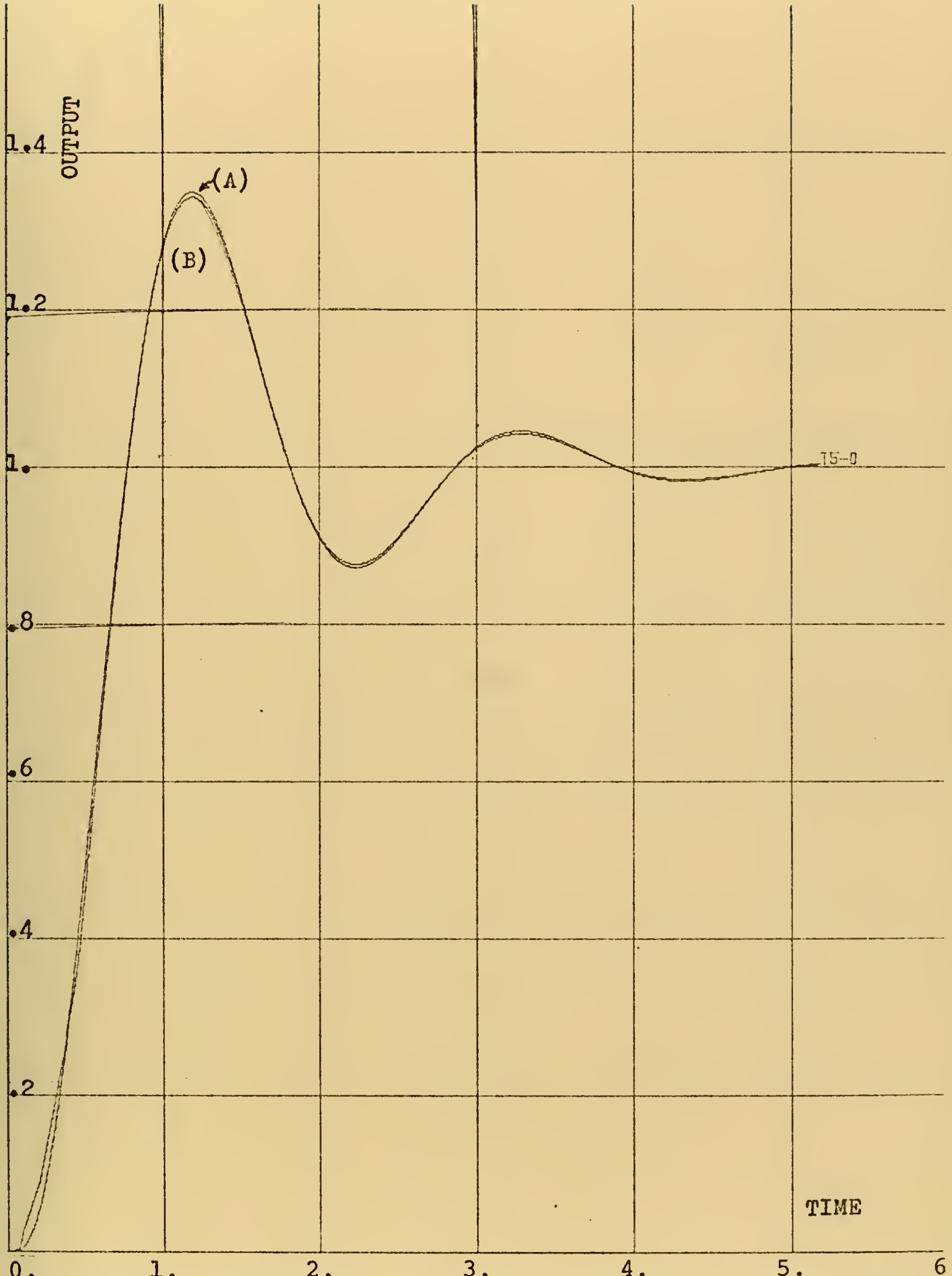

Figure III.8. EXAMPLE I. System's response (A) and the Five Poles no Zero model's response (B) to a unit step input.

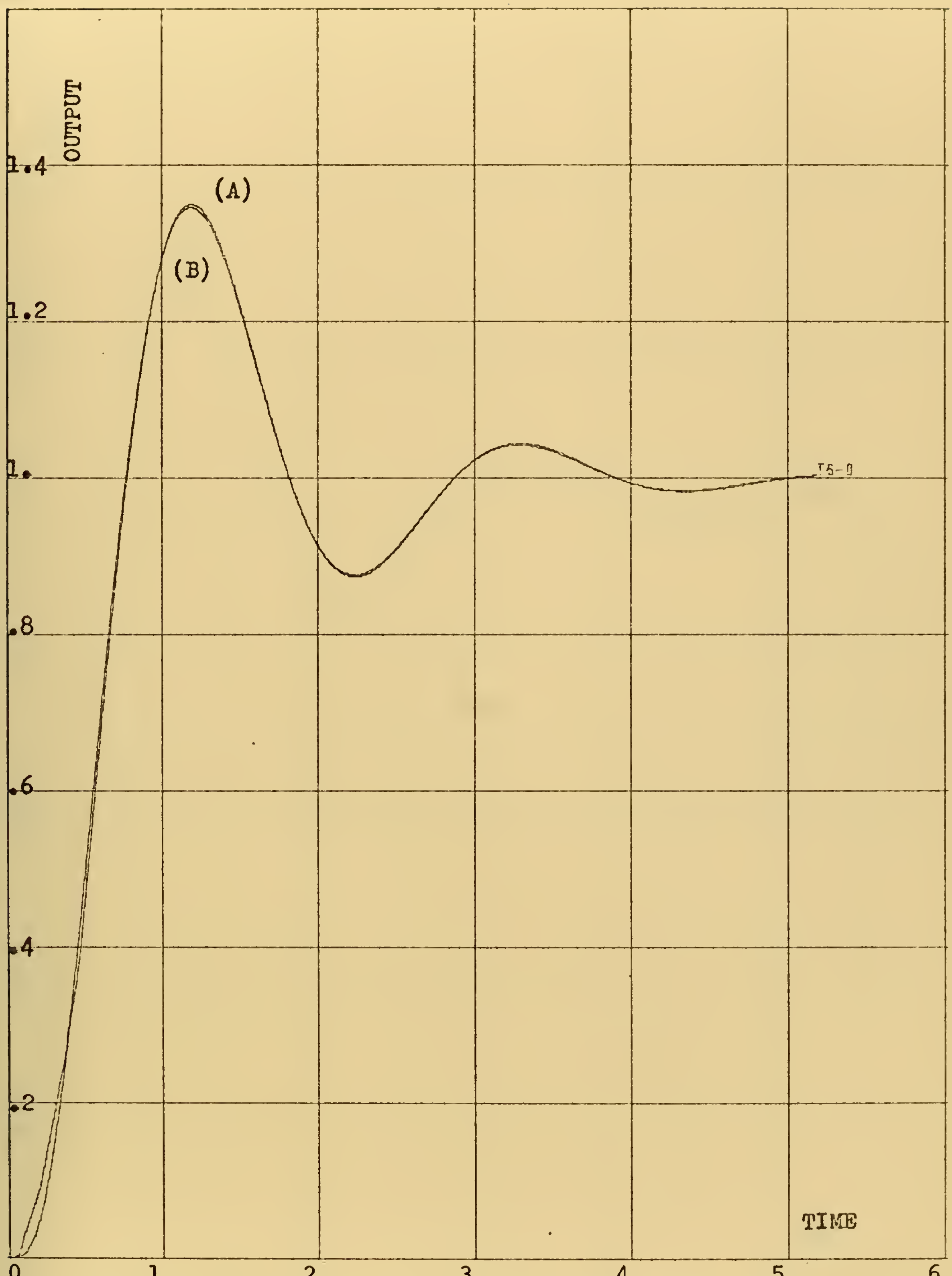

Figure III.9. EXAMPLE I. System's response (A) and the Six Poles no Zero model's response (B) to a unit step input.

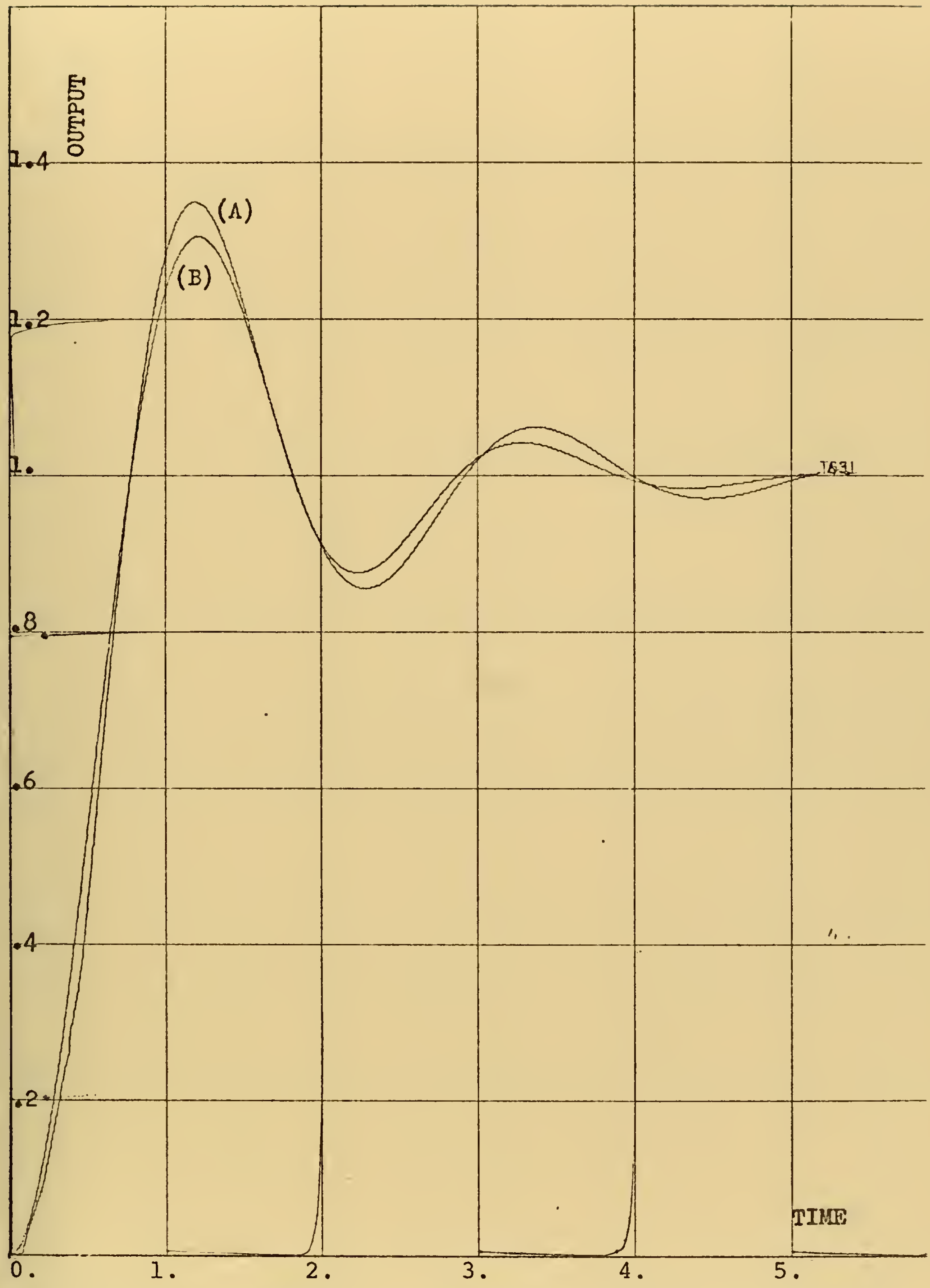

Figure III.10. EXAMPLE I. System's response (A) and the Three Poles and One Zero model's response (B) to a unit step input.

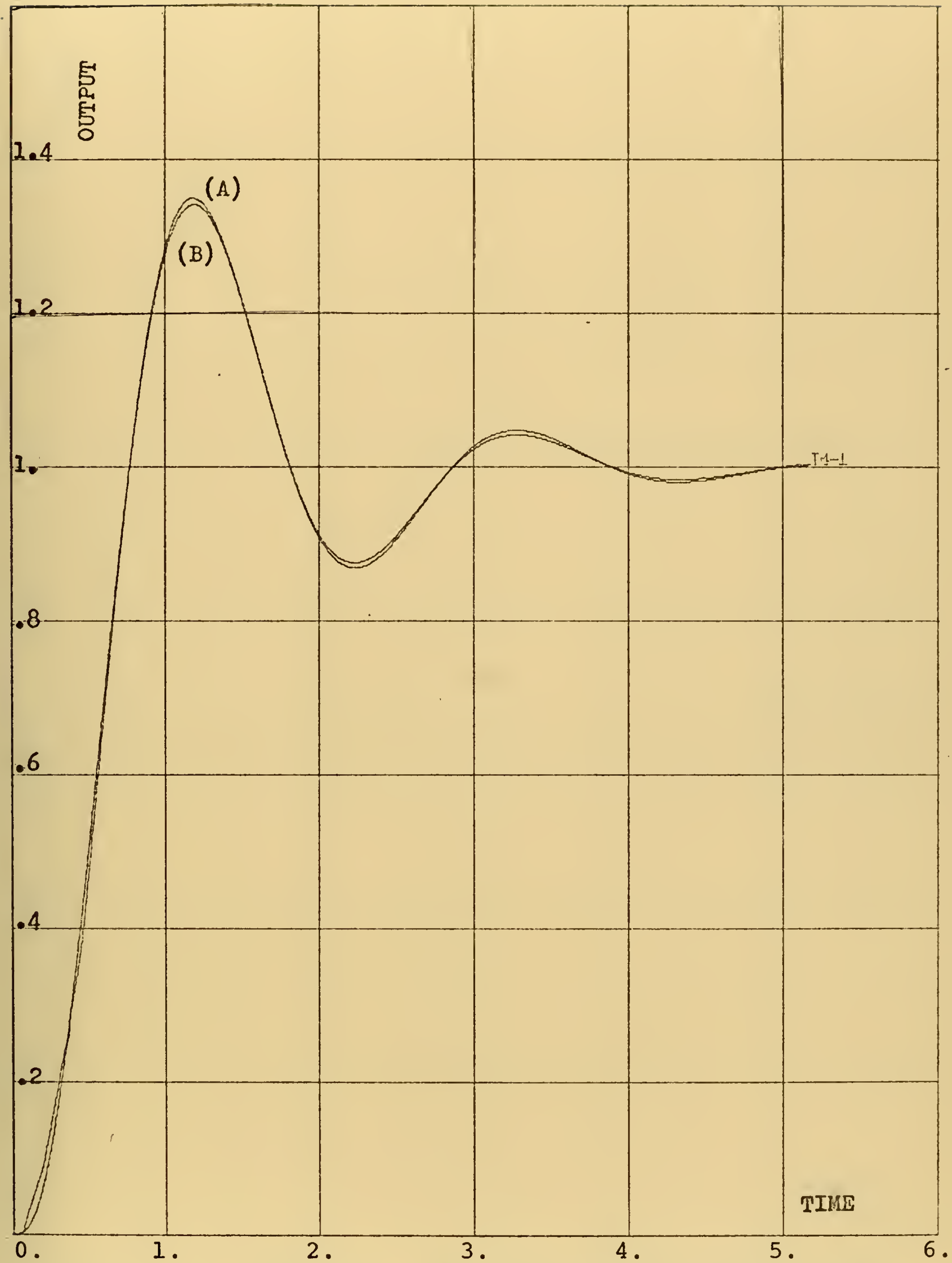

Figure III.11. EXAMPLE I. System's response (A) and the Four Poles and One Zero model's response (B) to a unit step input.

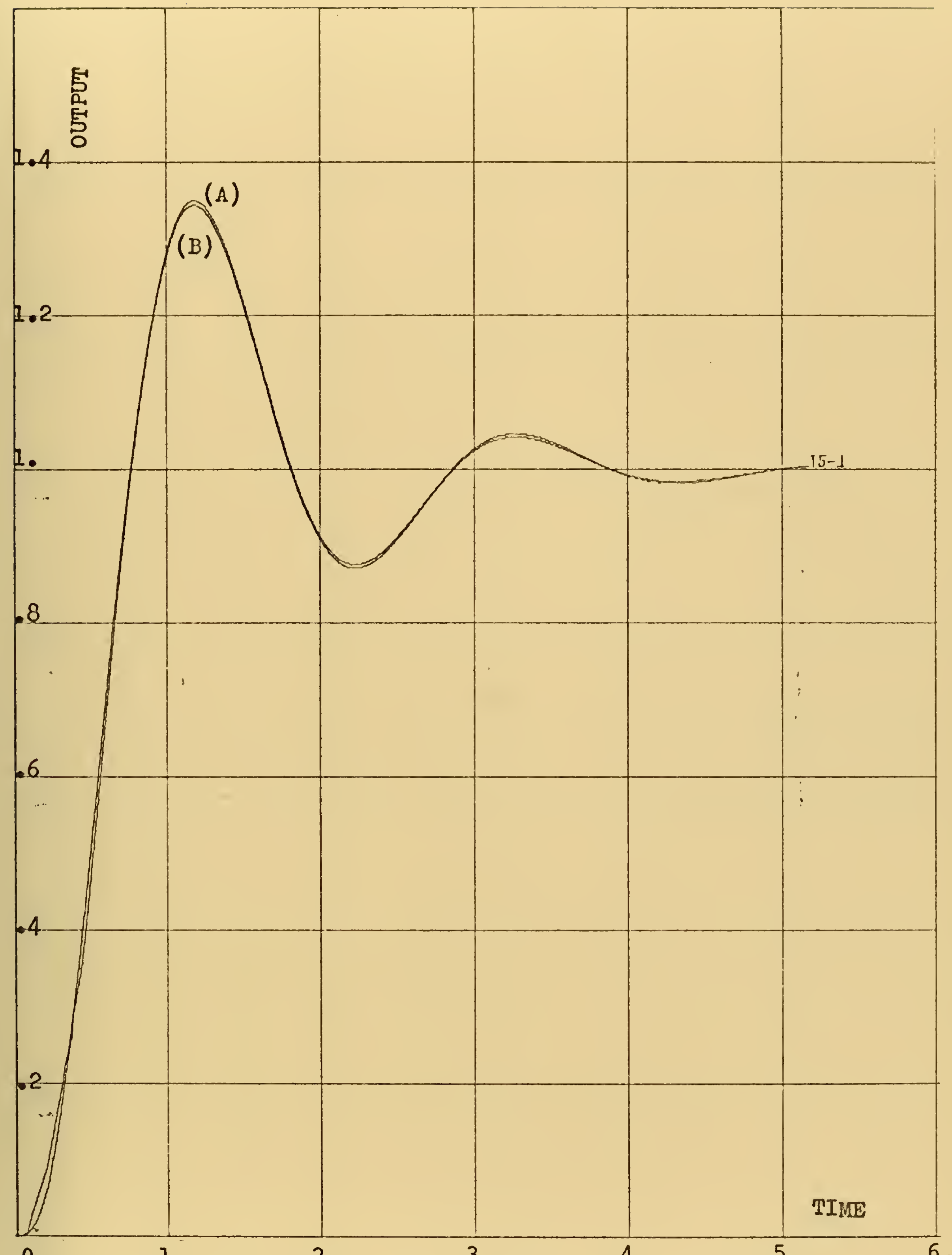

Figure III.12. EXAMPLE I. System's response (A) and the Five Poles and One Zero model's response (B) to a unit step input.

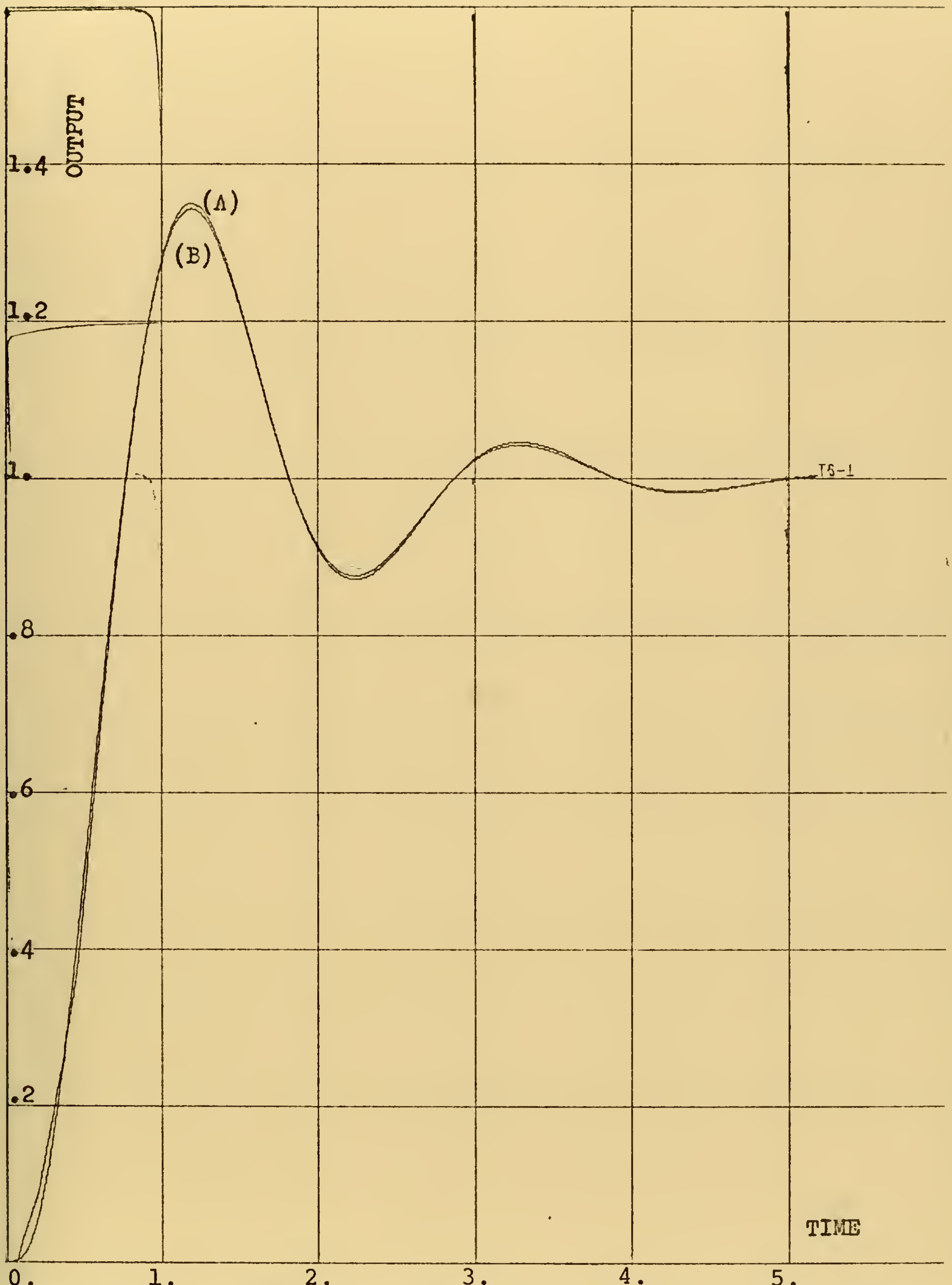

Figure III.13. EXAMPLE I. System's response (A) and the Six Poles and One Zero model's response (B) to a unit step input.

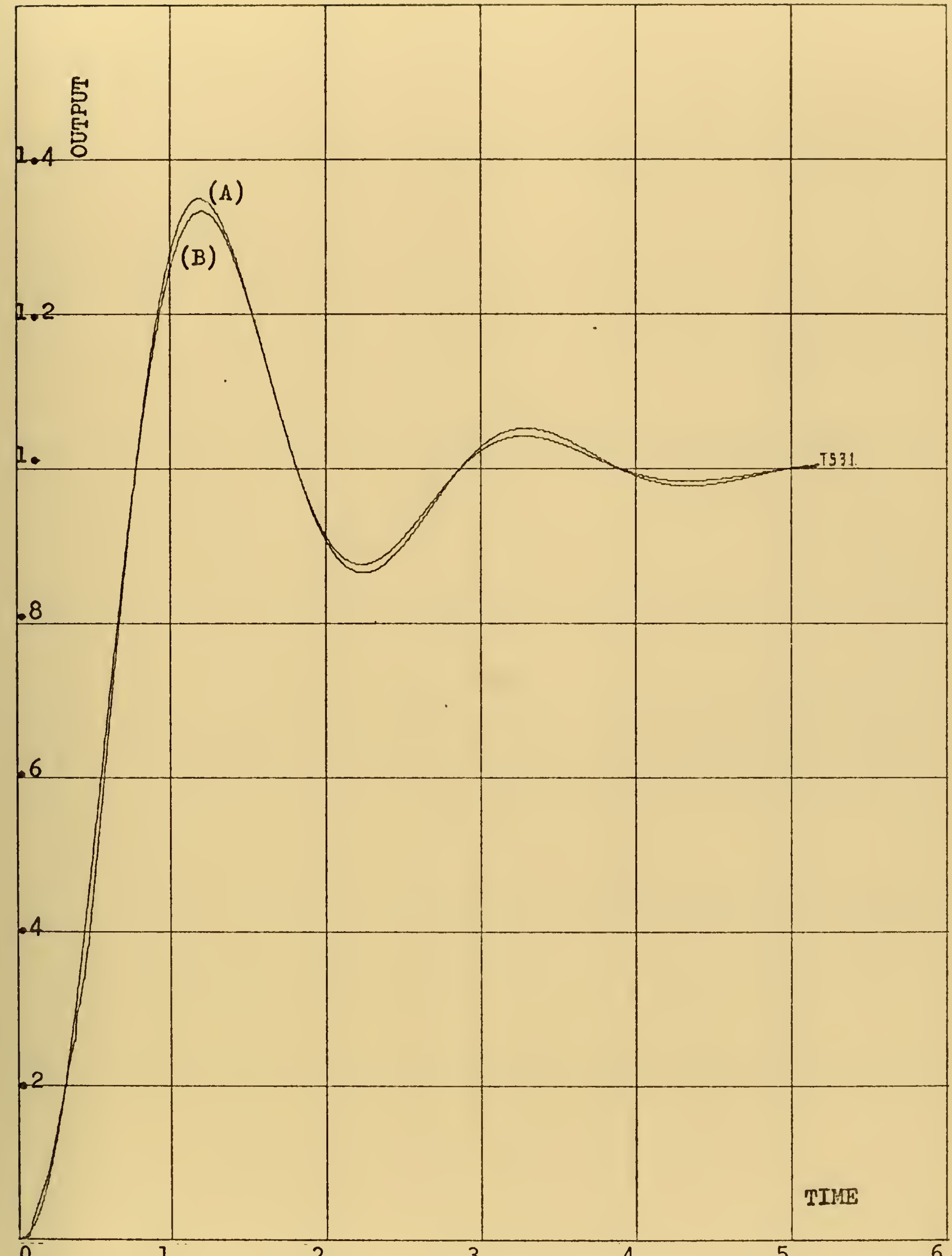

Figure III.14. EXAMPLE I. System's response (A) and the Three Poles and Two Zeros model's response (B) to a unit step input.

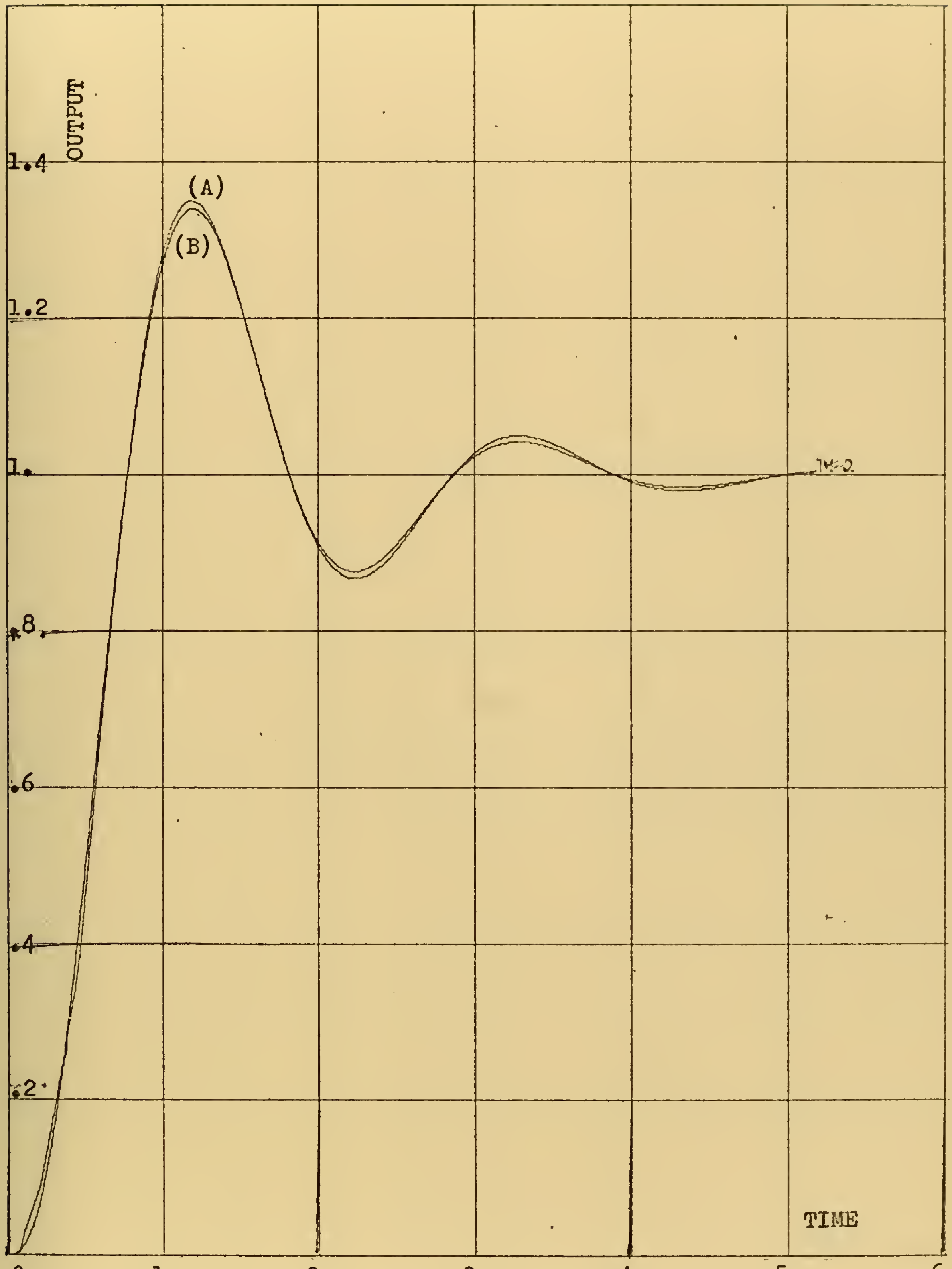

Figure III.15. EXAMPLE I. System's response (A) and the Four Poles and Two Zeros model's response (B) to a unit step input.

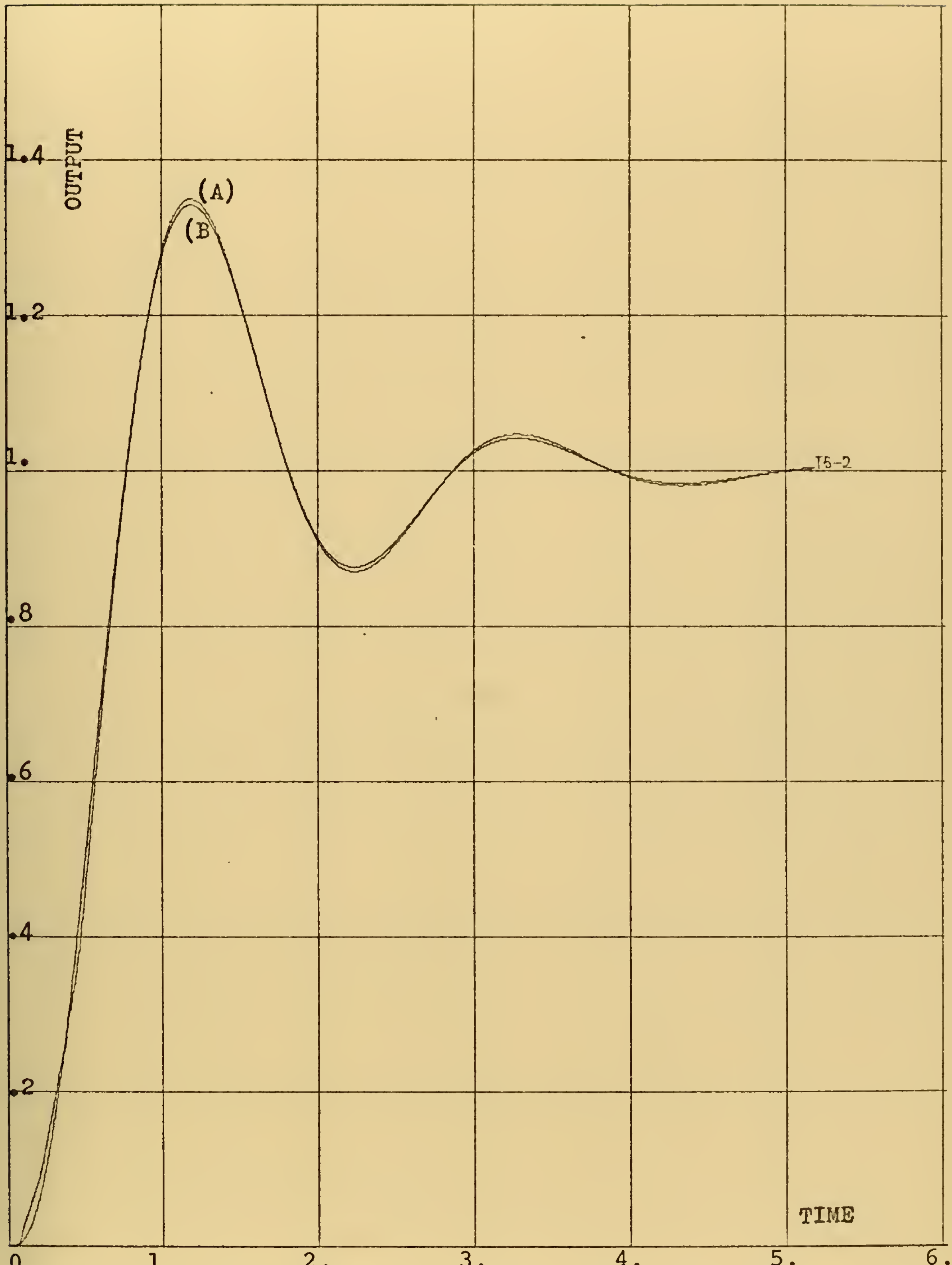

Figure III.16. EXAMPLE I. System's response (A) and the Six Poles and Two Zeros model's response (B) to a unit step input.

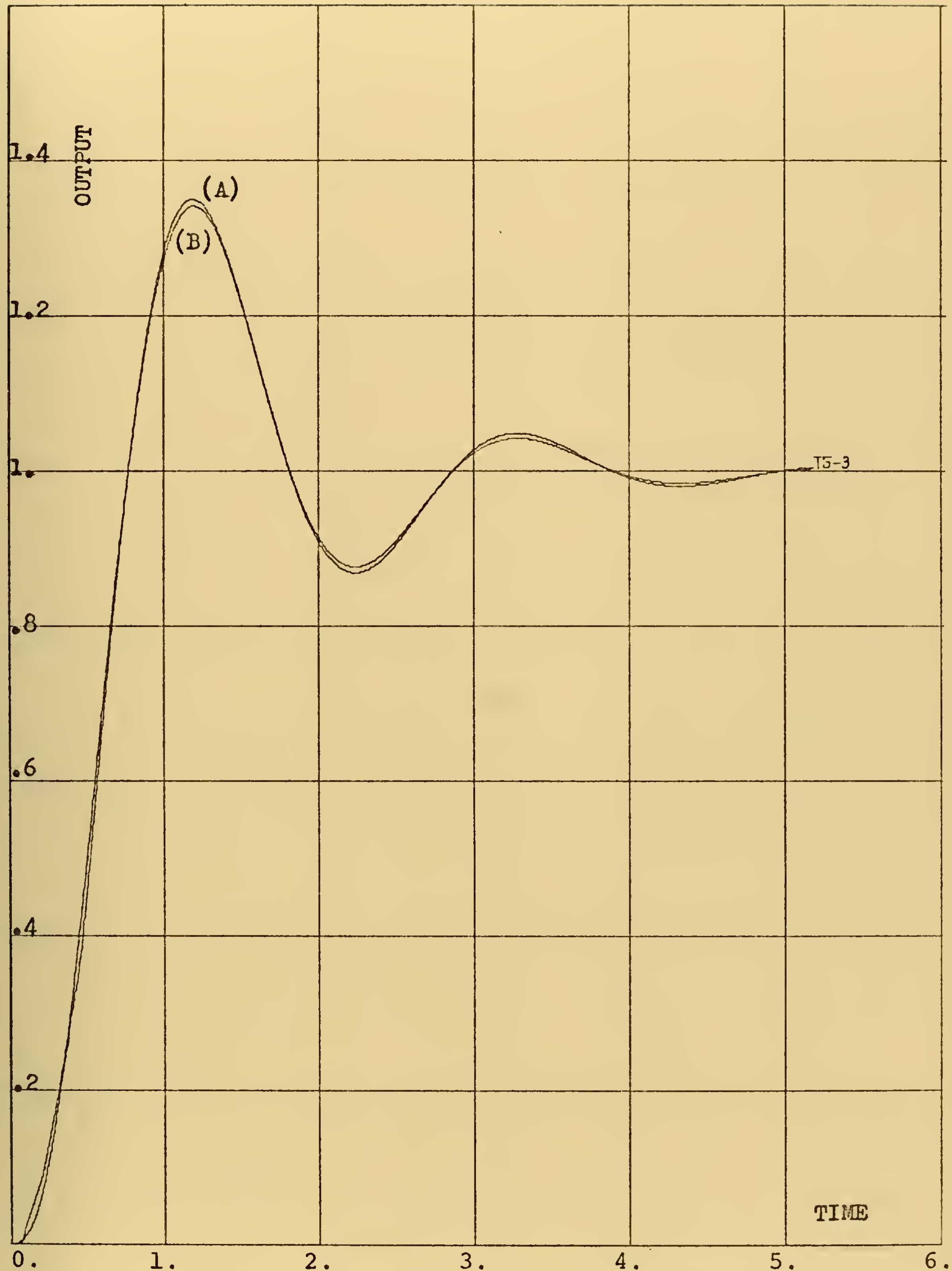

Figure III.17. EXAMPLE I. System's response (A) and the Five Poles and Three Zeros model's response (B) to a unit step input.

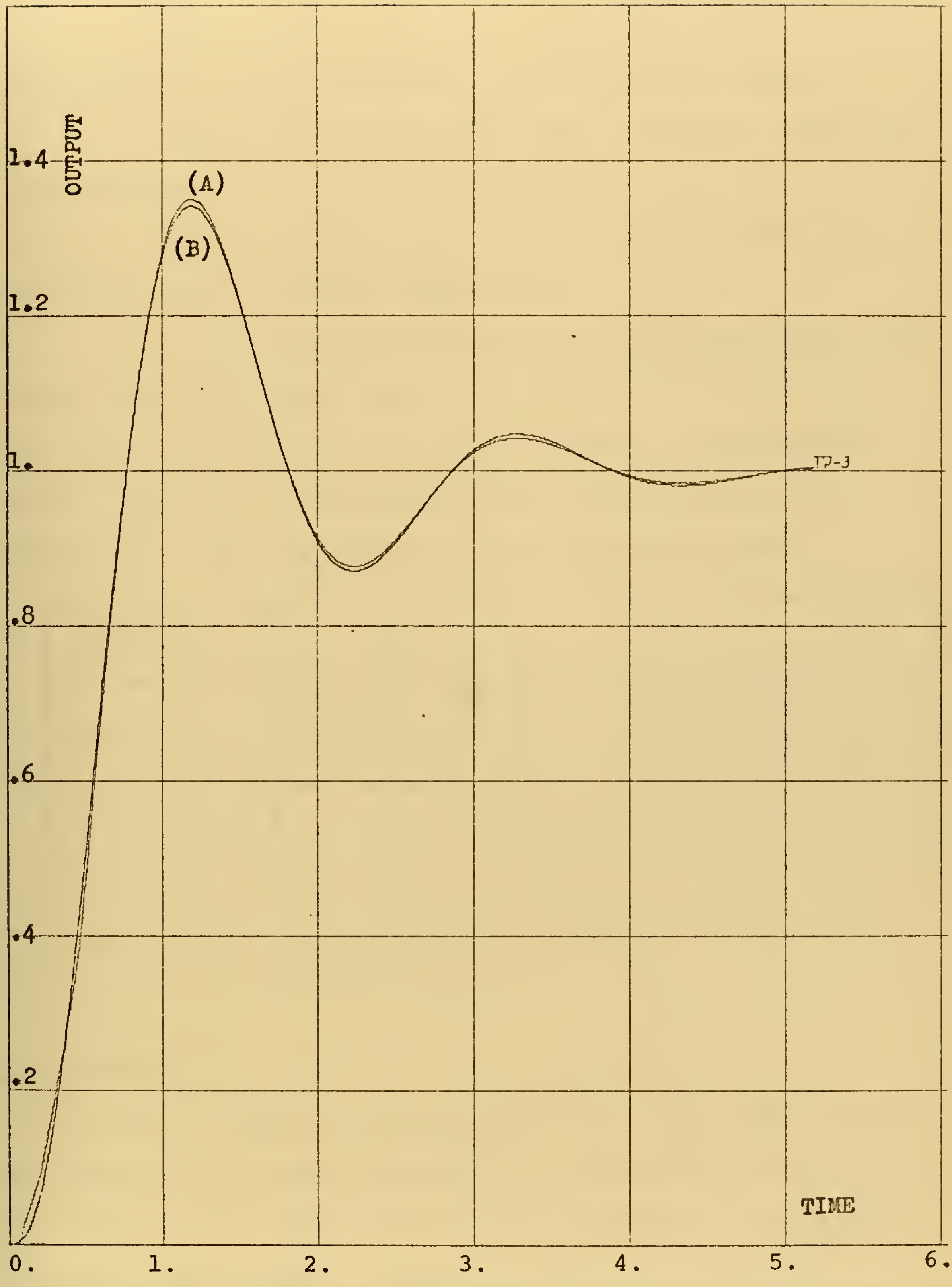

Figure III.18. EXAMPLE I. System's response (A) and the Seven Poles and Three Zeros model's response (B) to a unit step input.

a. A best optimum model seems feasible.

b. The models in the range from four poles to seven poles achieve a very acceptable error criterion value.

c. The second-order model does not seem a very good approximation.

d. The simplicity of a specific order model can be weighted versus criterion error value.

e. As the order of the model is increased the poles go far away on the real axis. On the other hand the error criterion value remains almost unchangeable increasing the order. Thus it can be thought that adding poles to the model is not going to minimize the performance index.

Zeros \ Poles	1	2	3	4	5	6	7
0		131.	5.17	4.7	5.09	6.09	6.4
1		140.	36.	4.4	4.8	5.0	5.4
2			8.4	4.5	4.5	4.61	4.8
3				4.24	4.402	4.406	4.47

TABLE III.5. Error criterion ($Jx10^4$) for models in Example I.

C. EXAMPLE II

The input-output measured data of the high-order system was taken as the step-response of a second-order system with the same features (Eq. (III.1)) as in Example I, delayed an interval approximately twice the above t_d. Thus, now $t_d' = 2t_d = 0.3$ seconds.

536 samples of the input-output assumed response (at intervals of 0.01 sec.) were used as measured data for the derivation of the l^{th}-order optimum models by minimization of the performance index defined in Eq. (III.6).

The investigation was performed using the technique of Example I. All possible models were determined of order ranging from no zeros and two poles to seven poles and three zeros. The optimum pole and zero locations obtained for each model are tabulated in Tables III.6 to III.9 and plots of the error criteria values versus number of poles are shown in Figs. III.19 and III.20 for specific numbers of zeros.

For further comparison between all models Table III.10 shows the performance index (error criteria) values for each. Figs. III.21 to III.36 show the time responses of various models and the original system.

1. Remarks

The following comments should be noted:

a. If $l>k$ the results show that the optimal model of order "l" is better than the optimal model of order "k". The error criterion value decrease as the order of the model increases, although in any cases the error values for the fourth-order model (no zeros) and higher can be considered acceptable.

b. For a specific number of poles, adding a zero to the model maintains the cost function value at about the same magnitude.

# of Poles	2	3	4	5	6	7
ζ	.405	.228	.255	.266	.275	.282
w_N	2.34	2.995	3.136	3.145	3.156	3.16
p_3		2.93	5.94	6.59	7.51	10.17
p_4			5.896	6.73	11.03	9.67
p_5				30.9	10.	10.52
p_6					174.91	68.89
p_7						70.33
$Jx10^4$	490.	63.6	18.5	12.6	8.2	6.3

TABLE III.6. EXAMPLE II. Optimum models with no zeros.

# of Poles	2	3	4	5	6	7
ζ	.407	.227	.252	.252	.274	.285
w_N	2.328	2.988	3.13	3.123	3.156	3.158
p_3		2.885	5.62	5.69	8.38	10.7
p_4			5.81	5.73	8.86	10.97
p_5				30.01	10.02	11.06
p_6					132.86	20.01
p_7						159.55
z_1	199.8	219.35	97.88	21.98	89.37	117.12
$Jx10^4$	500.	66.8	21.1	13.2	8.7	5.8

TABLE III.7. EXAMPLE II. Optimum models with one zero.

# of Poles	3	4	5	6	7
ζ	.222	.246	.257	.274	.277
w_N	2.965	3.105	3.13	3.154	3.154
p_3	2.75	5.13	5.14	6.96	7.03
p_4		5.69	10.37	9.998	11.17
p_5			12.64	13.77	13.48
p_6				29.8	28.8
p_7					101.97
z_1	64.98	57.85	40.2	88.64	61.9
z_2	199.38	75.6	149.5	102.87	163.5
$Jx10^4$	78.	26.9	15.4	8.4	7.9

TABLE III.8. EXAMPLE II. Optimum models with two zeros.

# of Poles	4	5	6	7
ζ	.244	.258	.271	
w_N	3.103	3.13	3.15	
p_3	5.064	5.53	9.001	
p_4	5.514	7.73	9.	
p_5		12.78	9.011	
p_6			20.	
z_1	49.12	38.52	27.65	
z_2	110.57	50.54	76.84	
z_3	119.32	191.07	175.3	
$Jx10^4$	29.1	17.3	9.3	7.9

TABLE III.9. EXAMPLE II. Optimum models with three zeros.

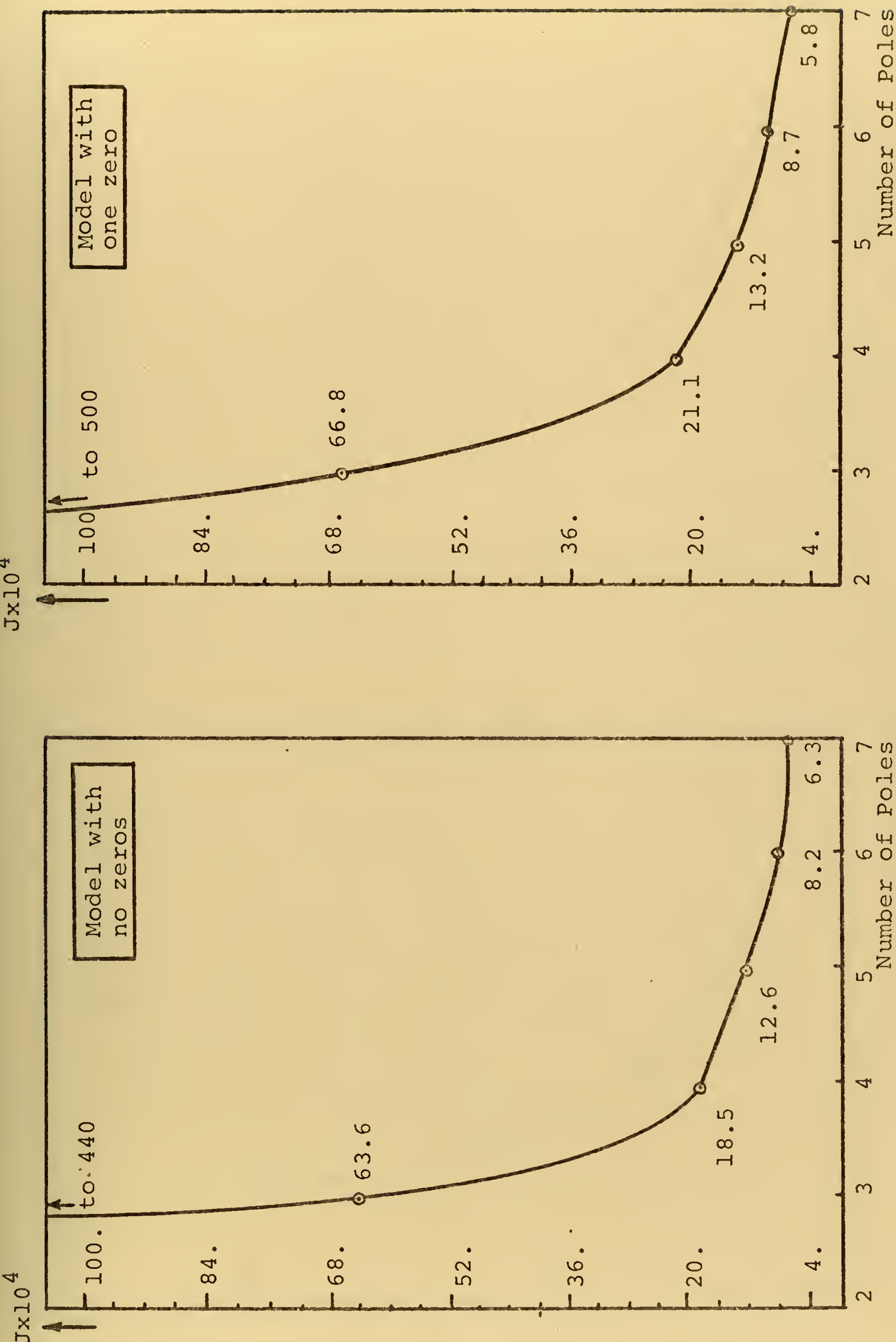

Figure III.19.

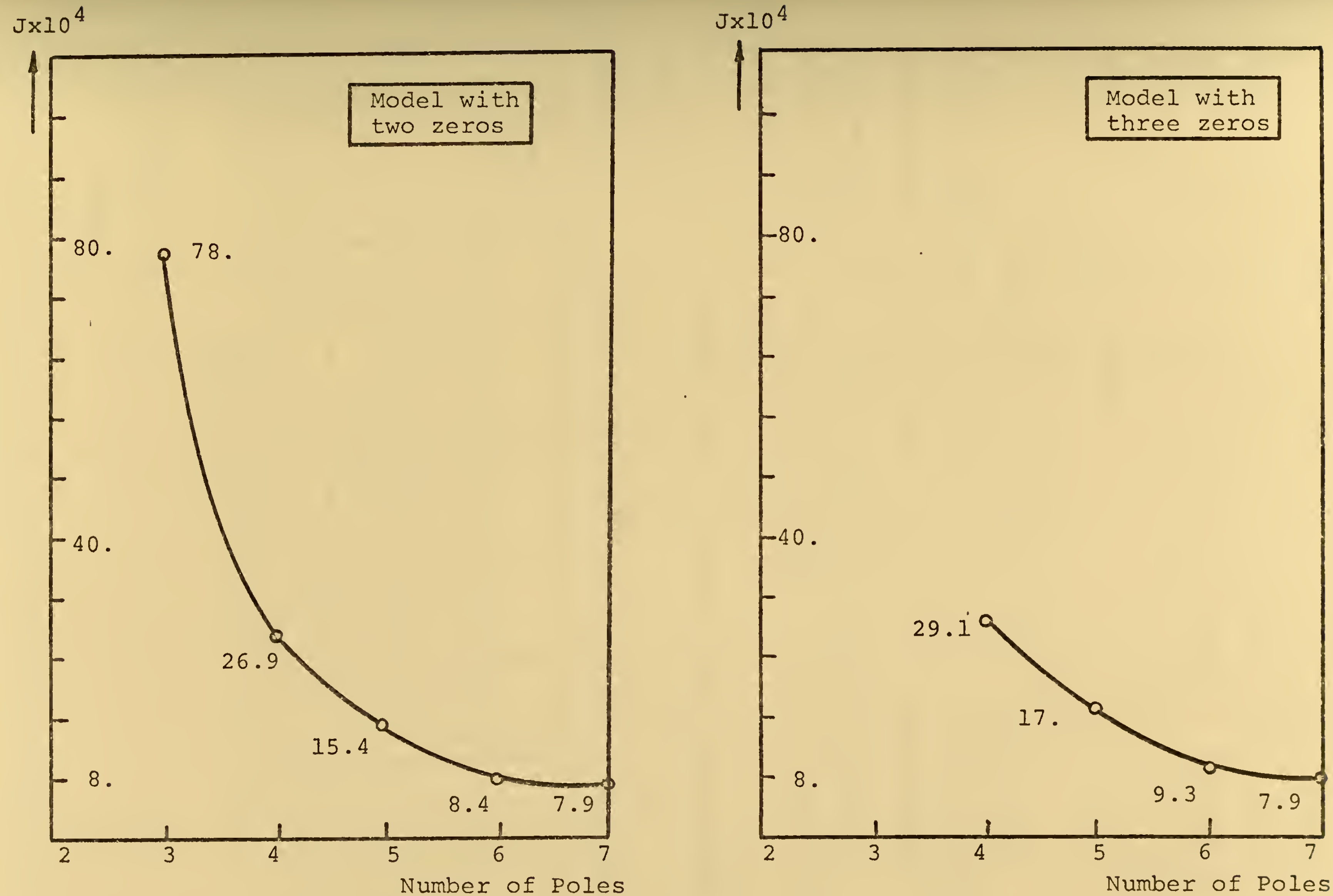

Figure III.20.

c. Between one model and the next one (increasing the order by adding a pole) the additional pole does not affect the initial pole location appreciably. They remain approximately the same, and the new pole is set far away to the left (in the s-plane).

d. The second-order model is the worst.

e. Selection of the desired model can be done easily depending on the desired simplicity, looking at the transients responses of both (original and approximated) systems and with a little margin for the cost function value it can be seen, in this example, that the fourth-order model fits very well.

f. The complex root values for the various models remain in a very small area, i.e., their values are essentially unchanged by the addition of poles and zeros.

Zeros \ Poles	1	2	3	4	5	6	7
0		490.	63.6	18.5	12.6	8.2	6.3
1		500.	66.8	21.1	13.2	8.7	5.8
2			78.	26.9	15.4	8.4	7.9
3				29.1	17.	9.3	7.9

TABLE III.10. Error criterion ($J \times 10^4$) for models in Example II.

D. EXAMPLE III

1. General

A transfer function G(s) of a seventh-order system with no finite zeros is given by

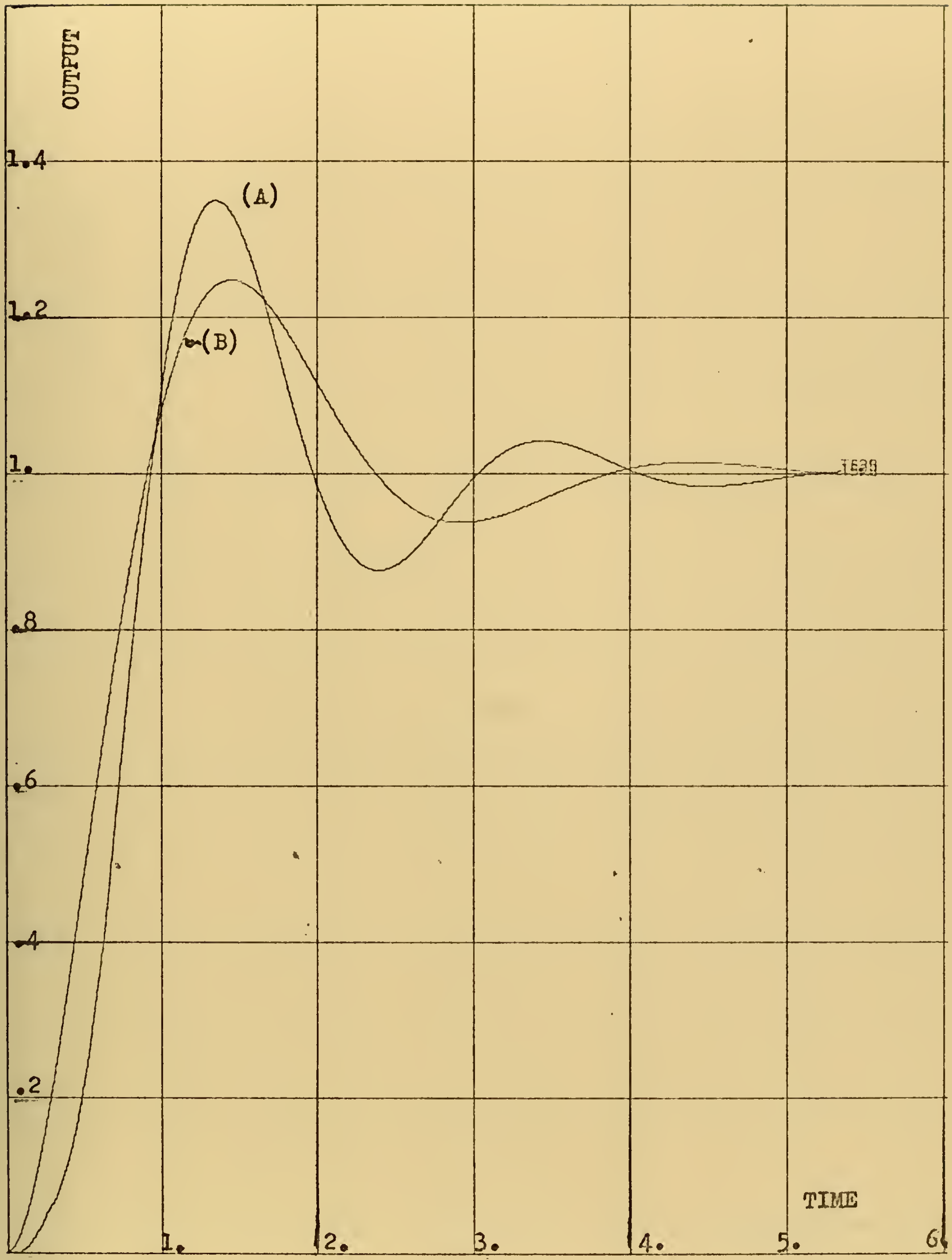

Figure III.21. EXAMPLE II. System's response (A) and the Two Poles no Zero model's response (B) to a unit step input.

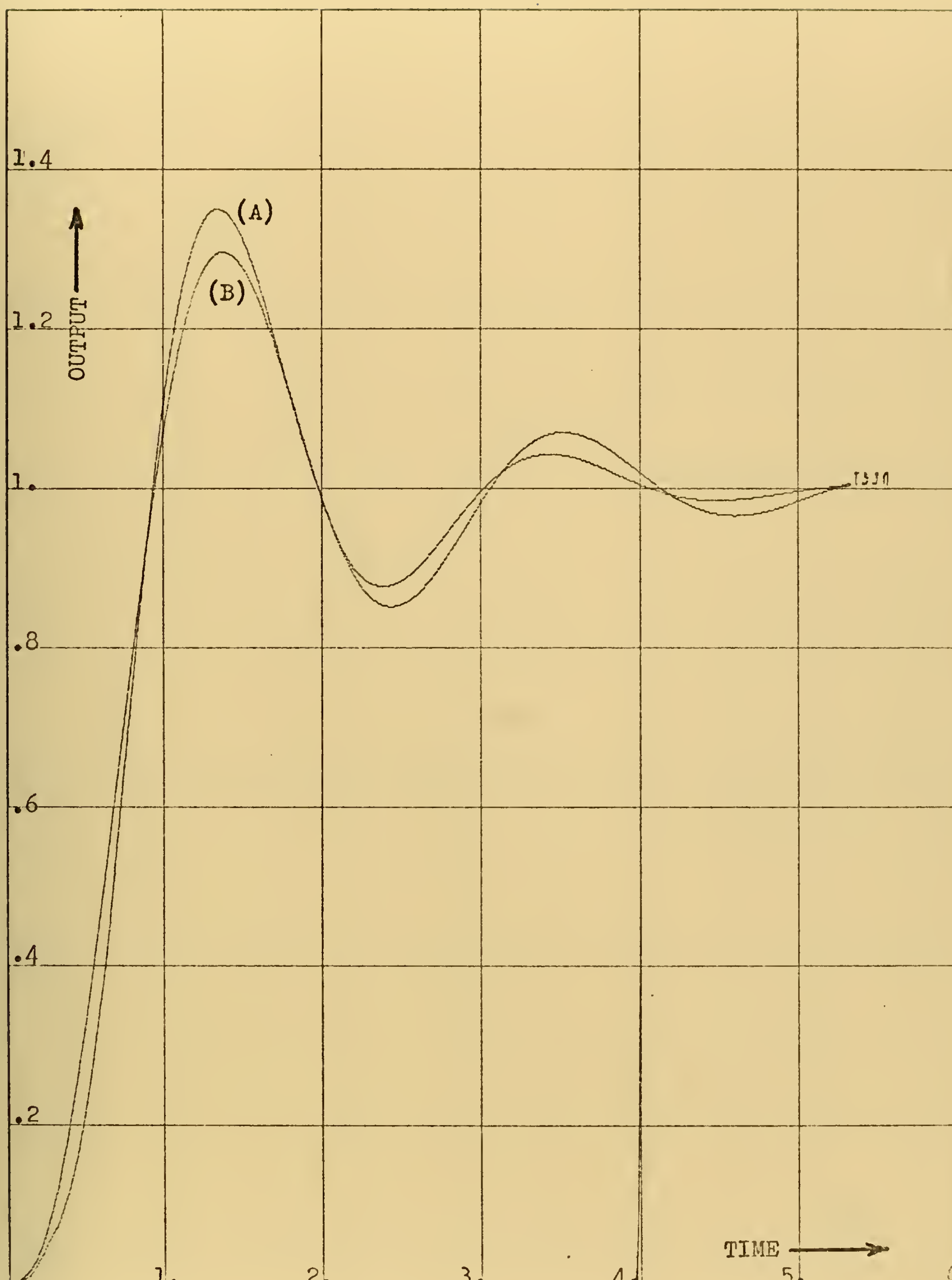

Figure III.22. EXAMPLE II. System's response (A) and the Three Poles no Zero model's response (B) to a unit step input.

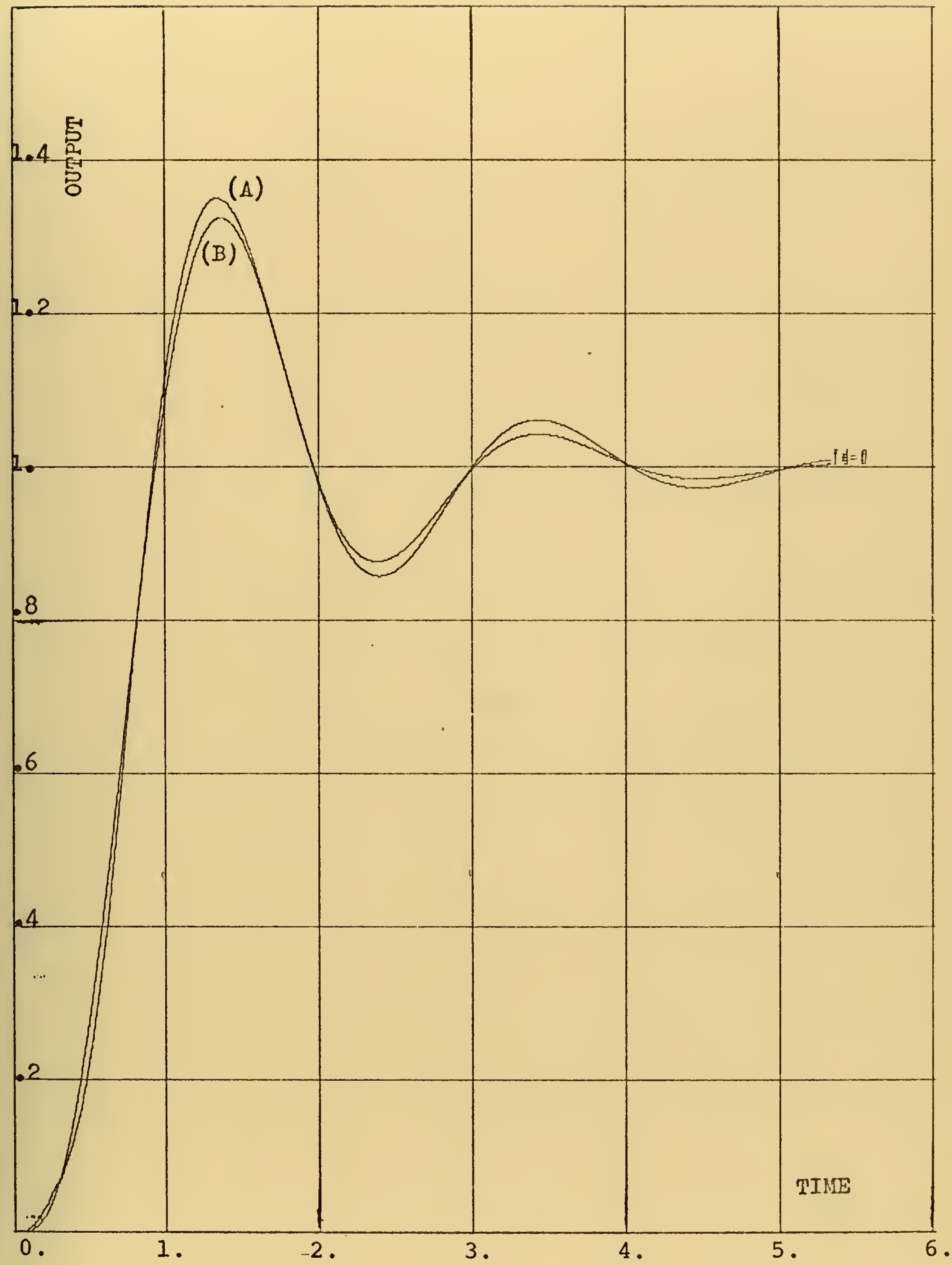

Figure III.23. EXAMPLE II. System's response (A) and the Four Poles no Zero model's response (B) to a unit step input.

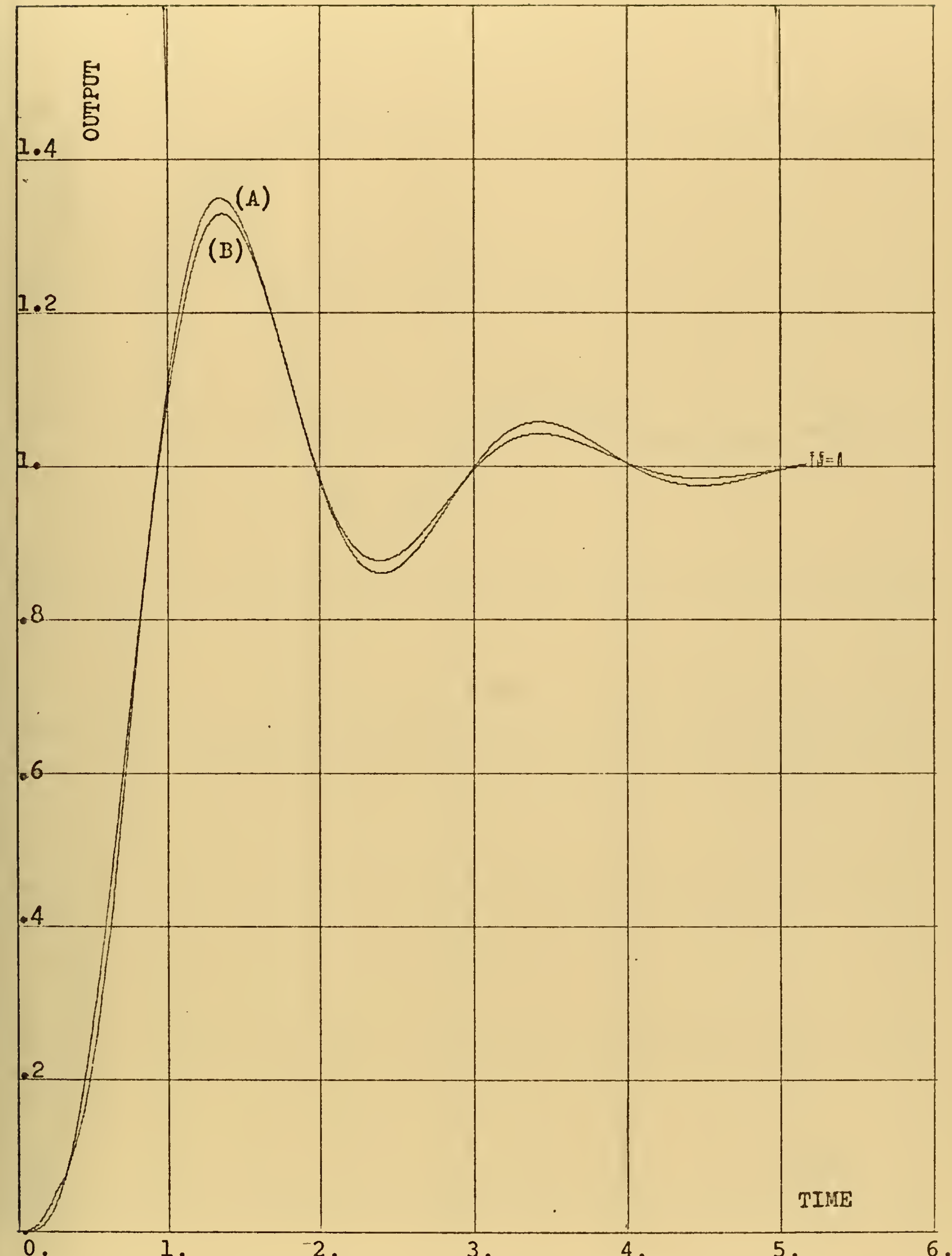

Figure III.24. EXAMPLE II. System's response (A) and the Five Poles no Zero model's response (B) to a unit step input.

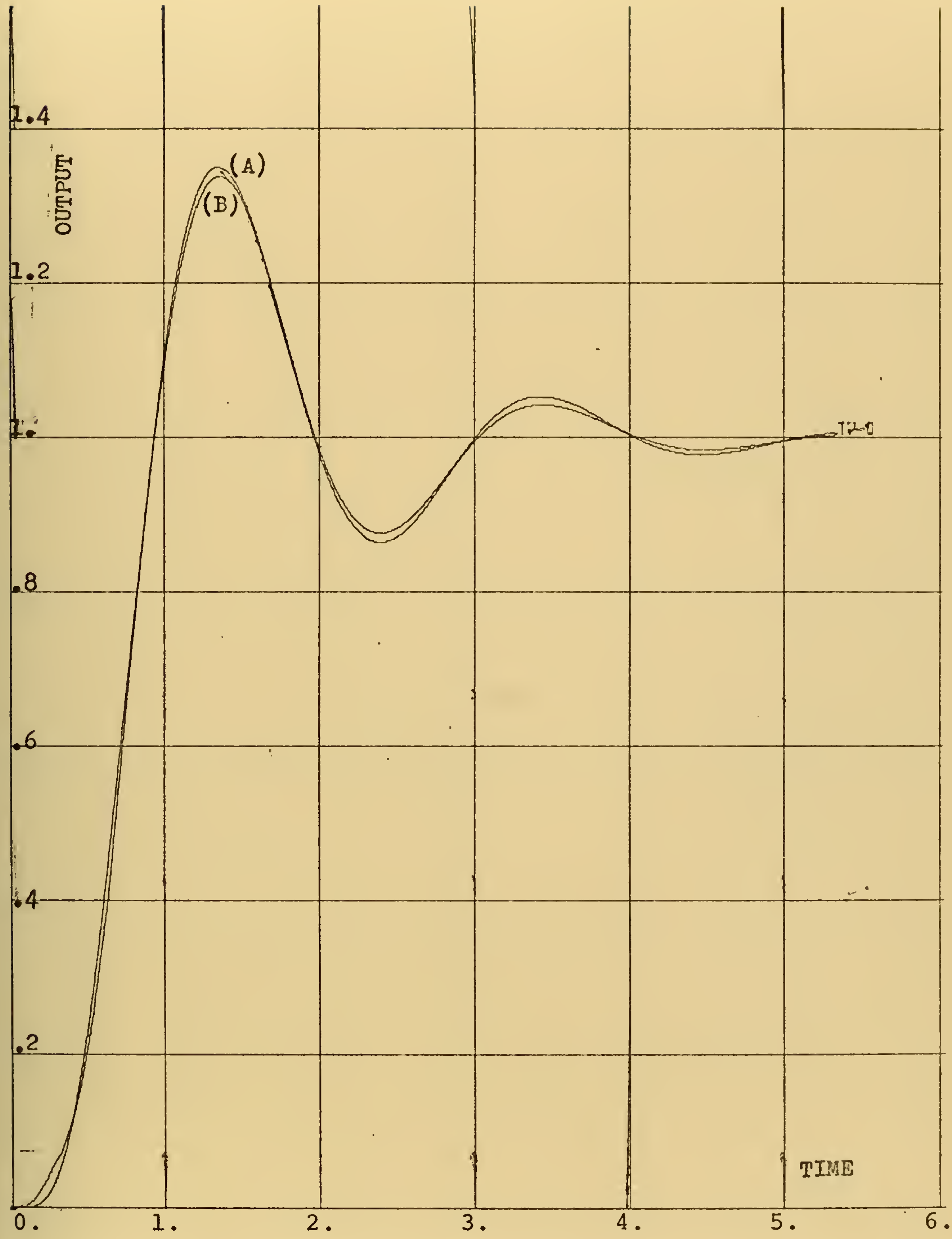

Figure III.25. EXAMPLE II. System's response (A) and the Seven Poles no Zero model's response (B) to a unit step input.

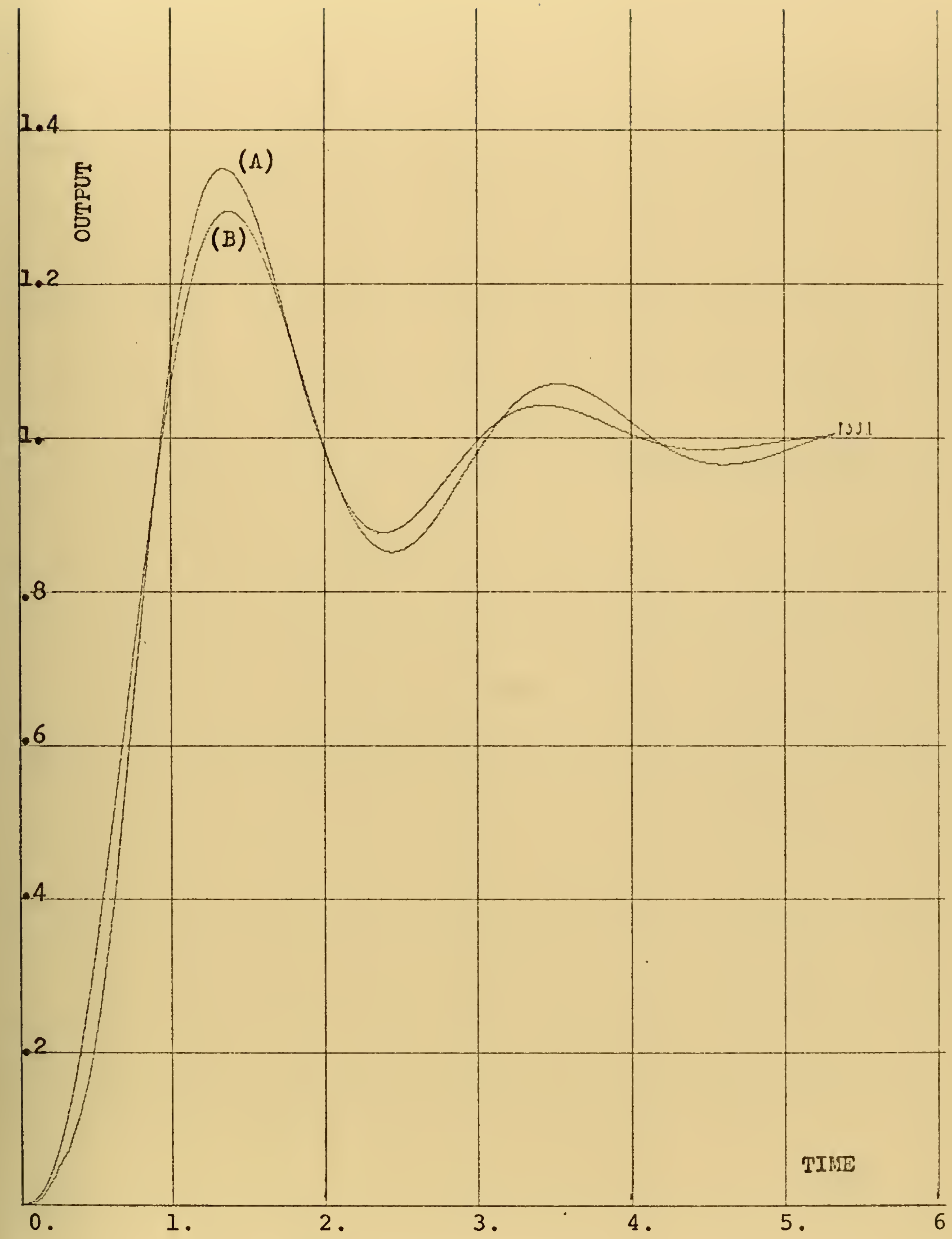

Figure III.26. EXAMPLE II. System's response (A) and the Three Poles and One Zero model's response (B) to a unit step input.

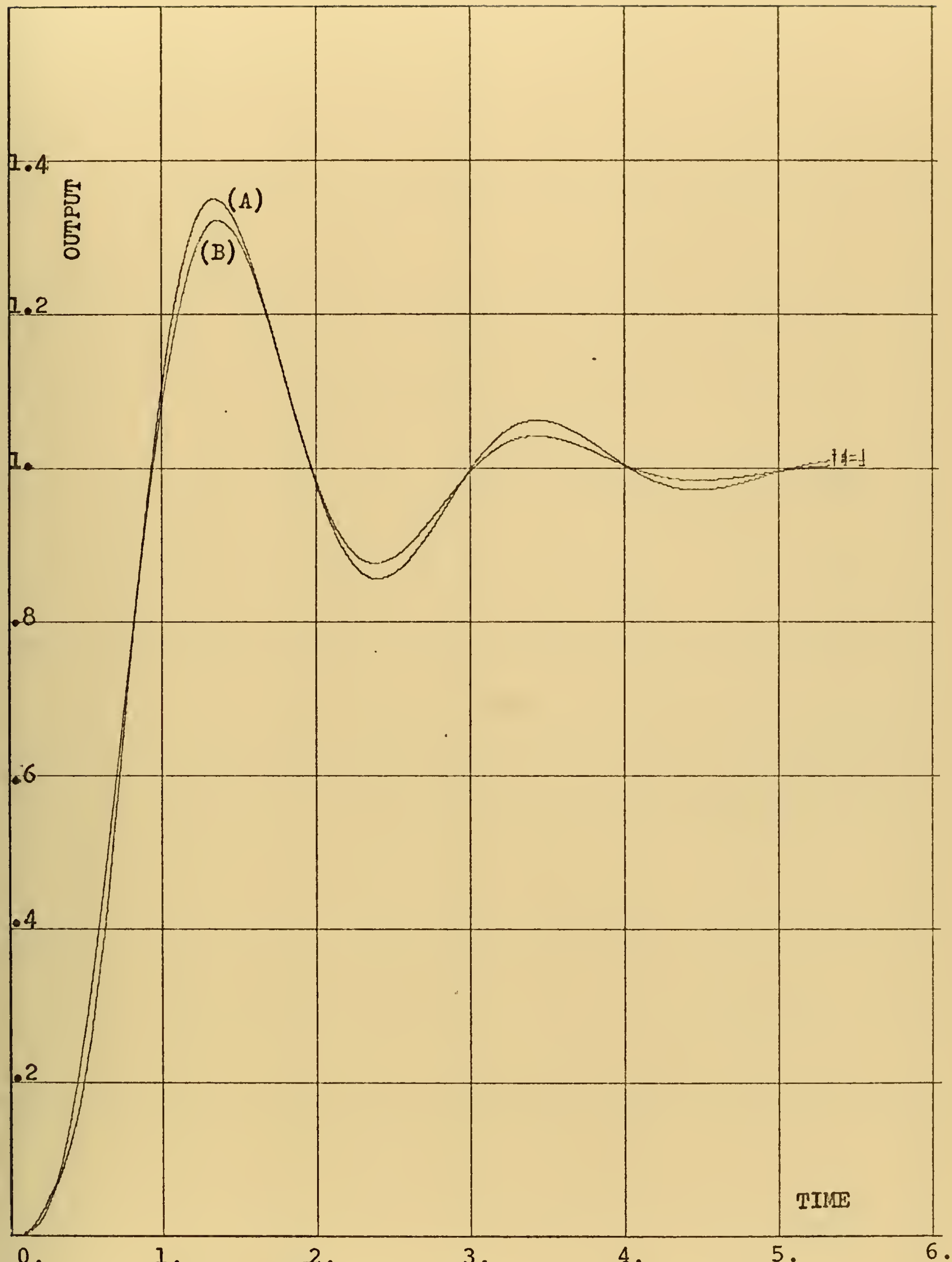

Figure III.27. EXAMPLE II. System's response (A) and the Four Poles and One Zero model's response (B) to a unit step input.

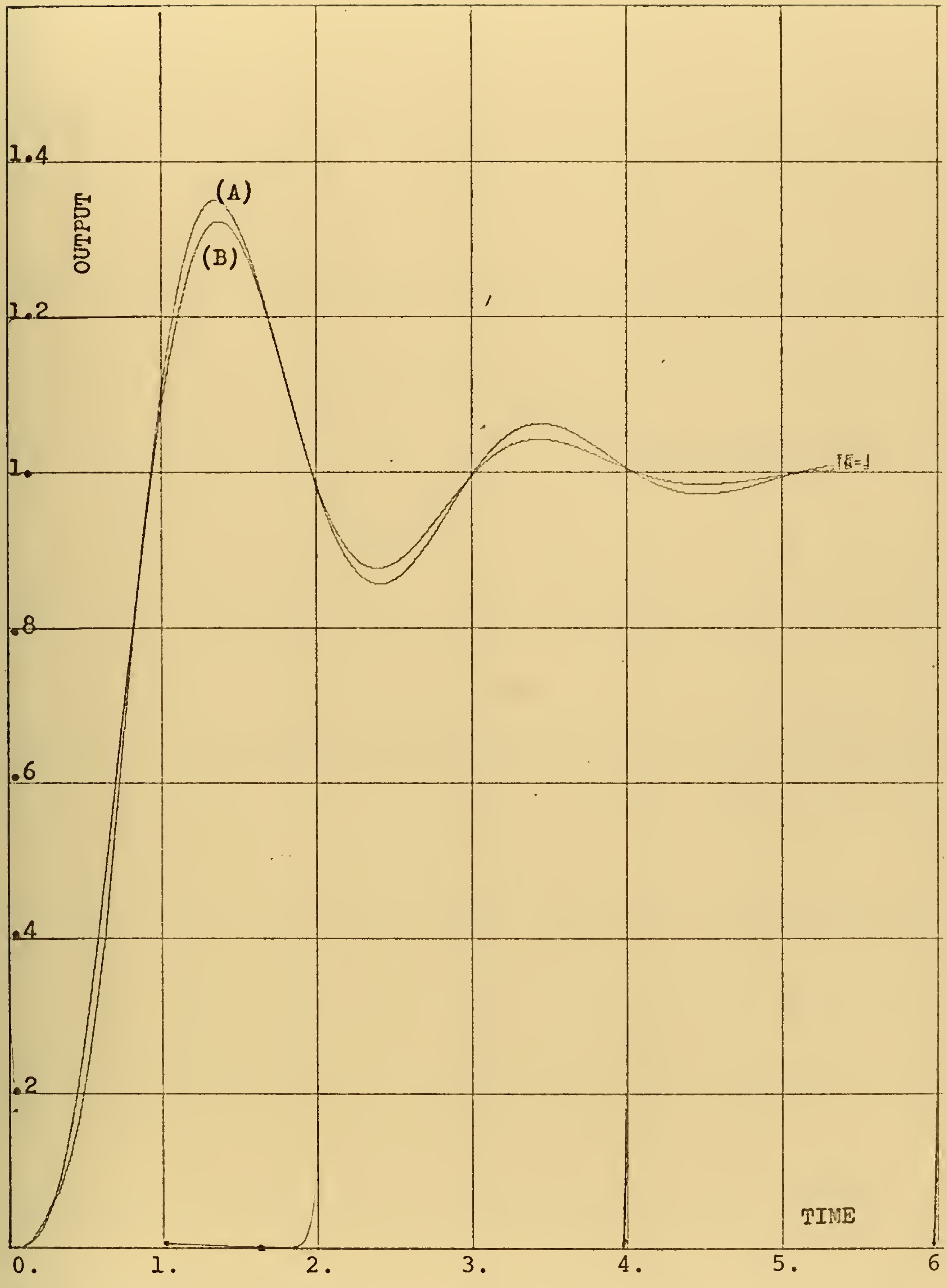

Figure III.28. EXAMPLE II. System's response (A) and the Five Poles and One Zero model's response (B) to a unit step input.

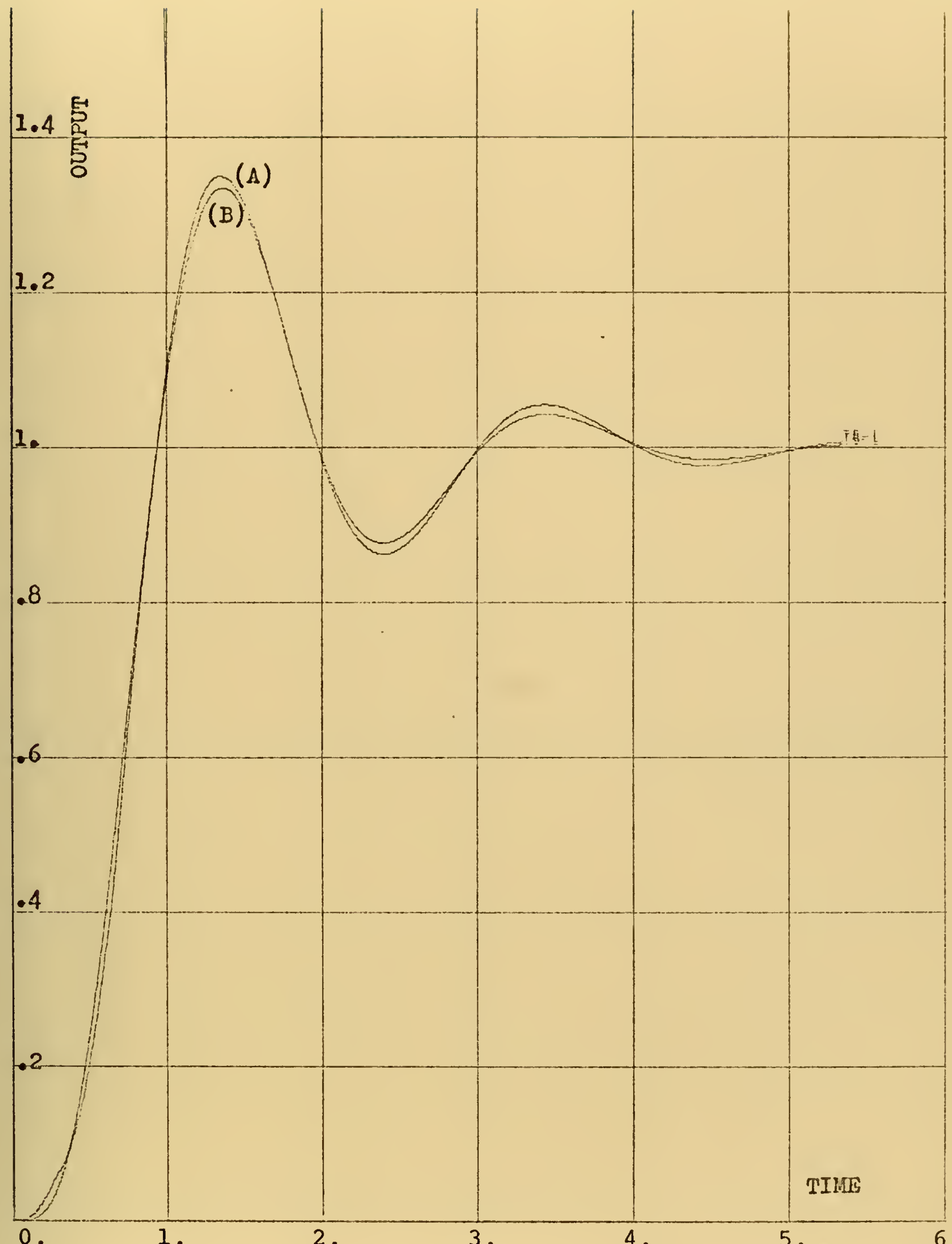

Figure III.29. EXAMPLE II. System's response (A) and the Six Poles and One Zero model's response (B) to a unit step input.

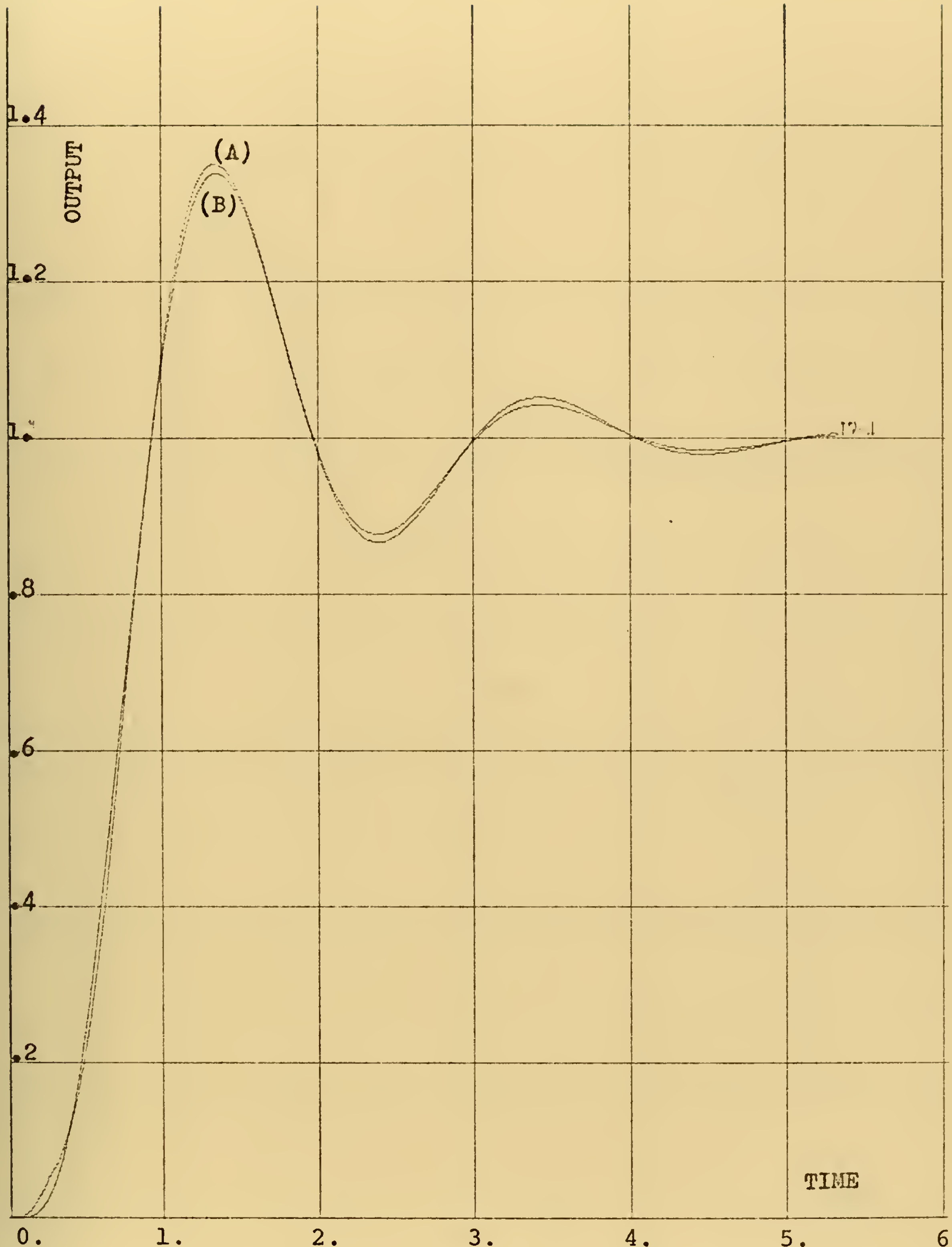

Figure III.30. EXAMPLE II. System's response (A) and the Seven Poles and One Zero model's response (B) to a unit step input.

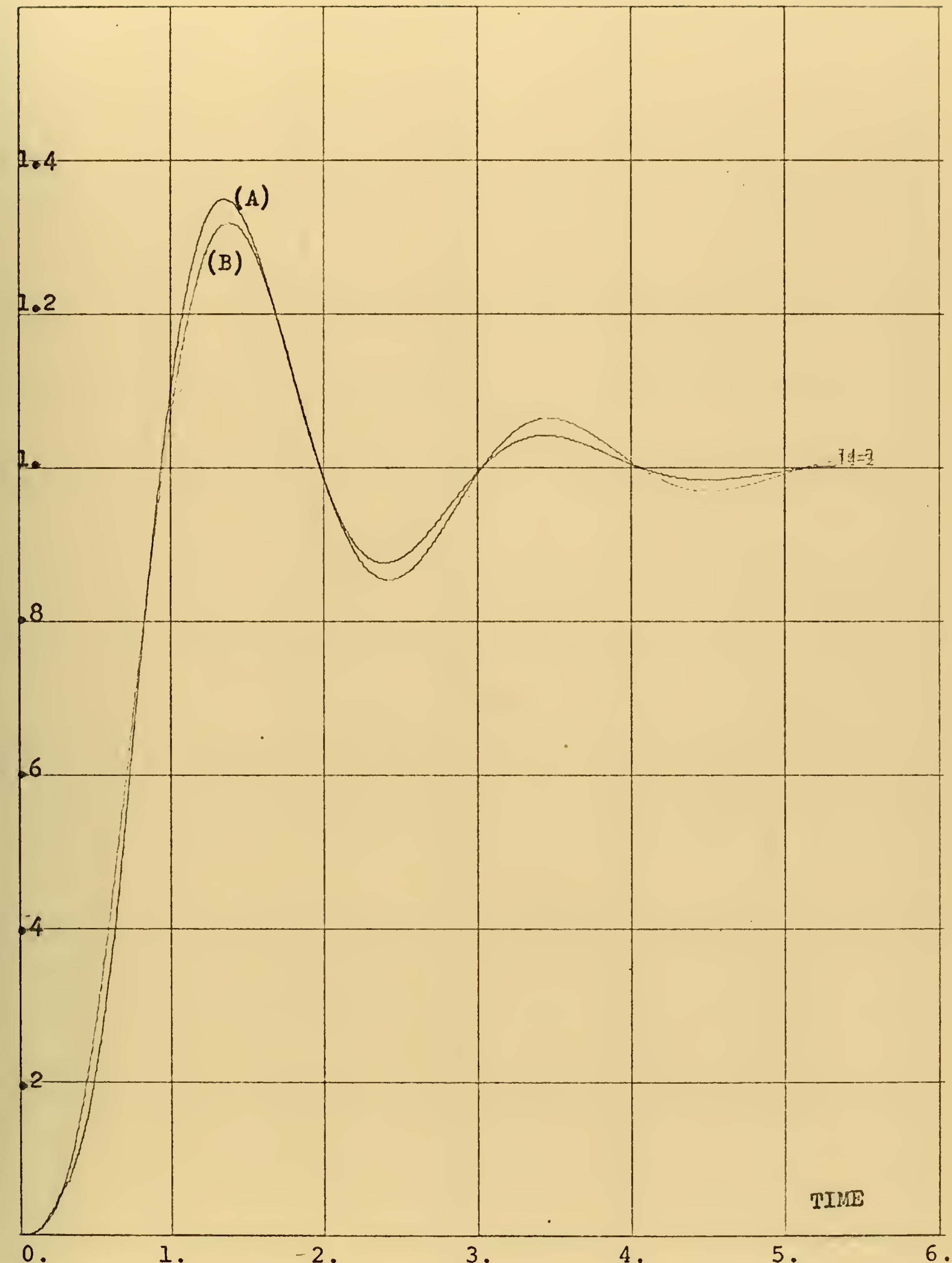

Figure III.31. EXAMPLE II. System's response (A) and the Four Poles and Two Zeros model's response (B) to a unit step input.

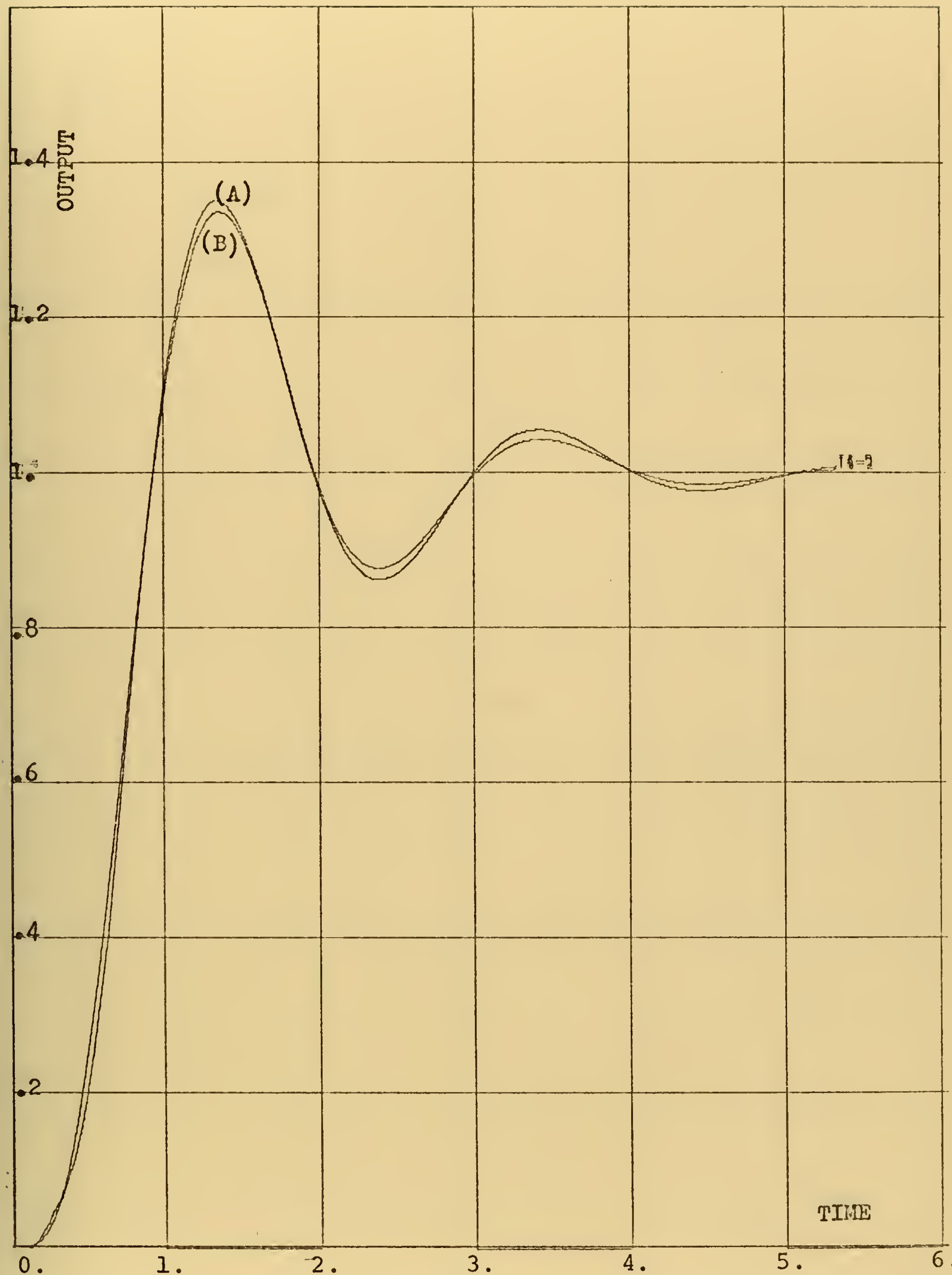

Figure III.32. EXAMPLE II. System's response (A) and the Six Poles and Two Zeros model's response (B) to a unit step input.

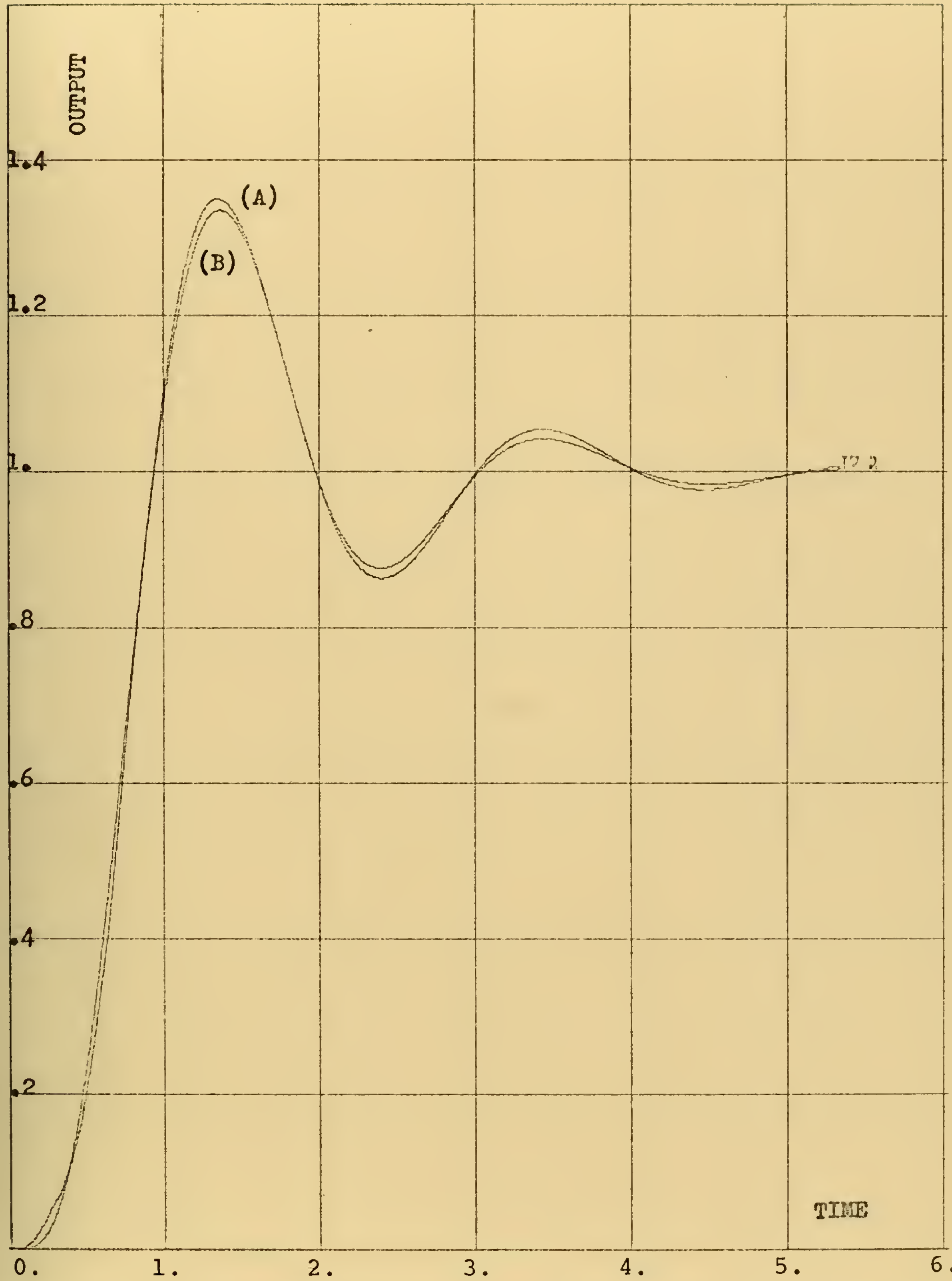

Figure III.33. EXAMPLE II. System's response (A) and the Seven Poles and Two Zeros model's response (B) to a unit step input.

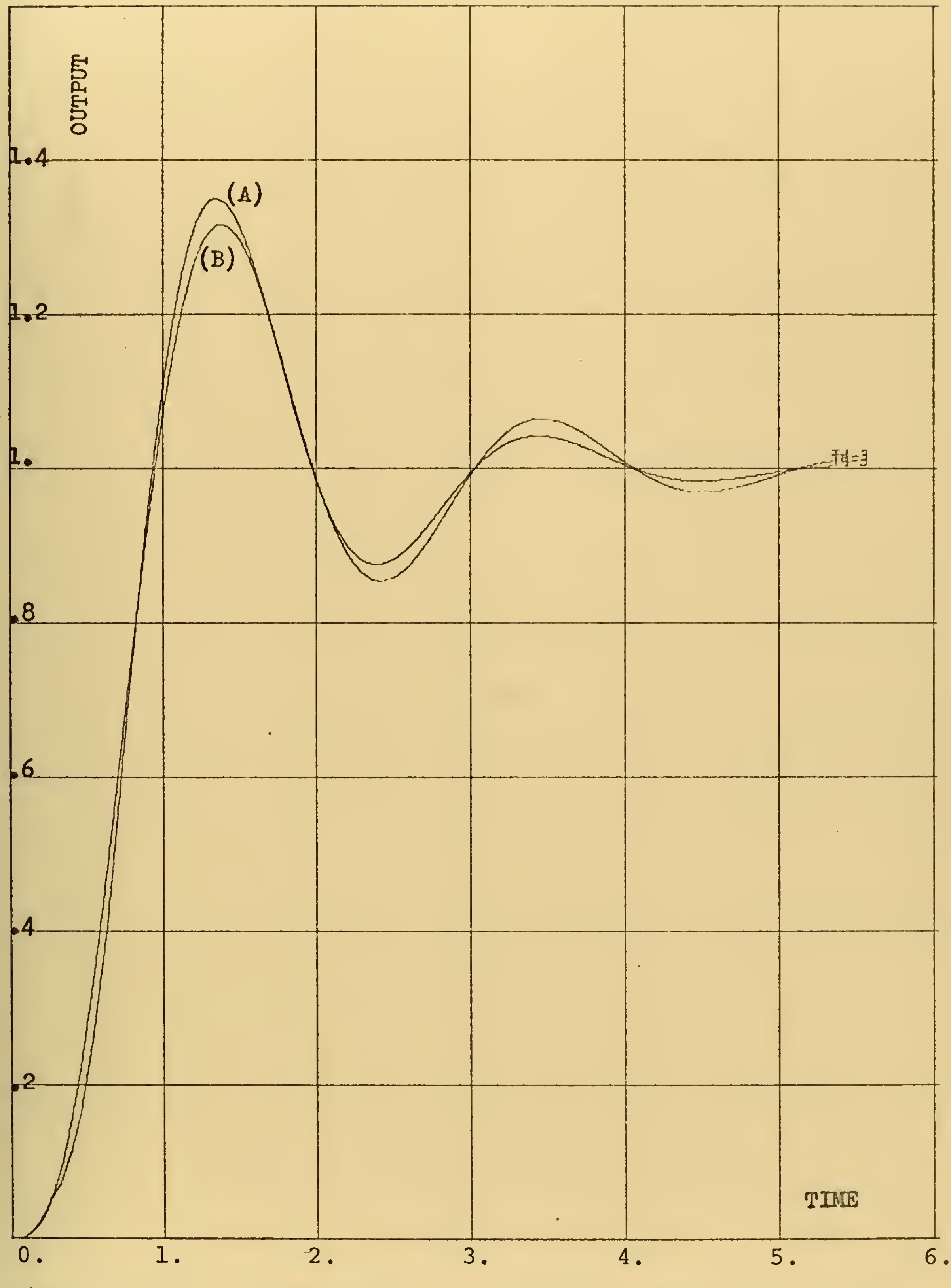

Figure III.34. EXAMPLE II. System's response (A) and the Four Poles and Three Zeros model's response (B) to a unit step input.

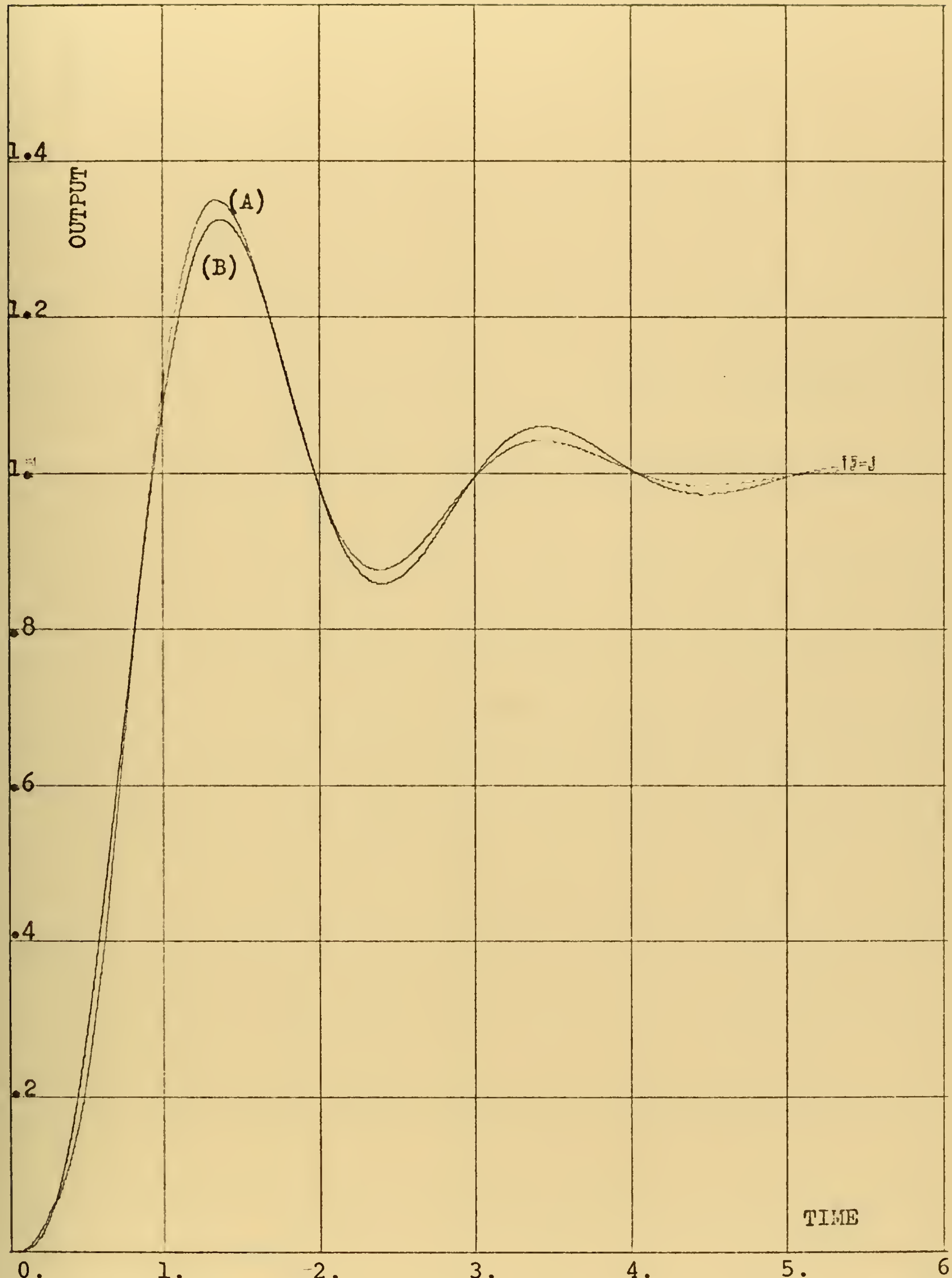

Figure III.35. EXAMPLE II. System's response (A) and the Five Poles and Three Zeros model's response (B) to a unit step input.

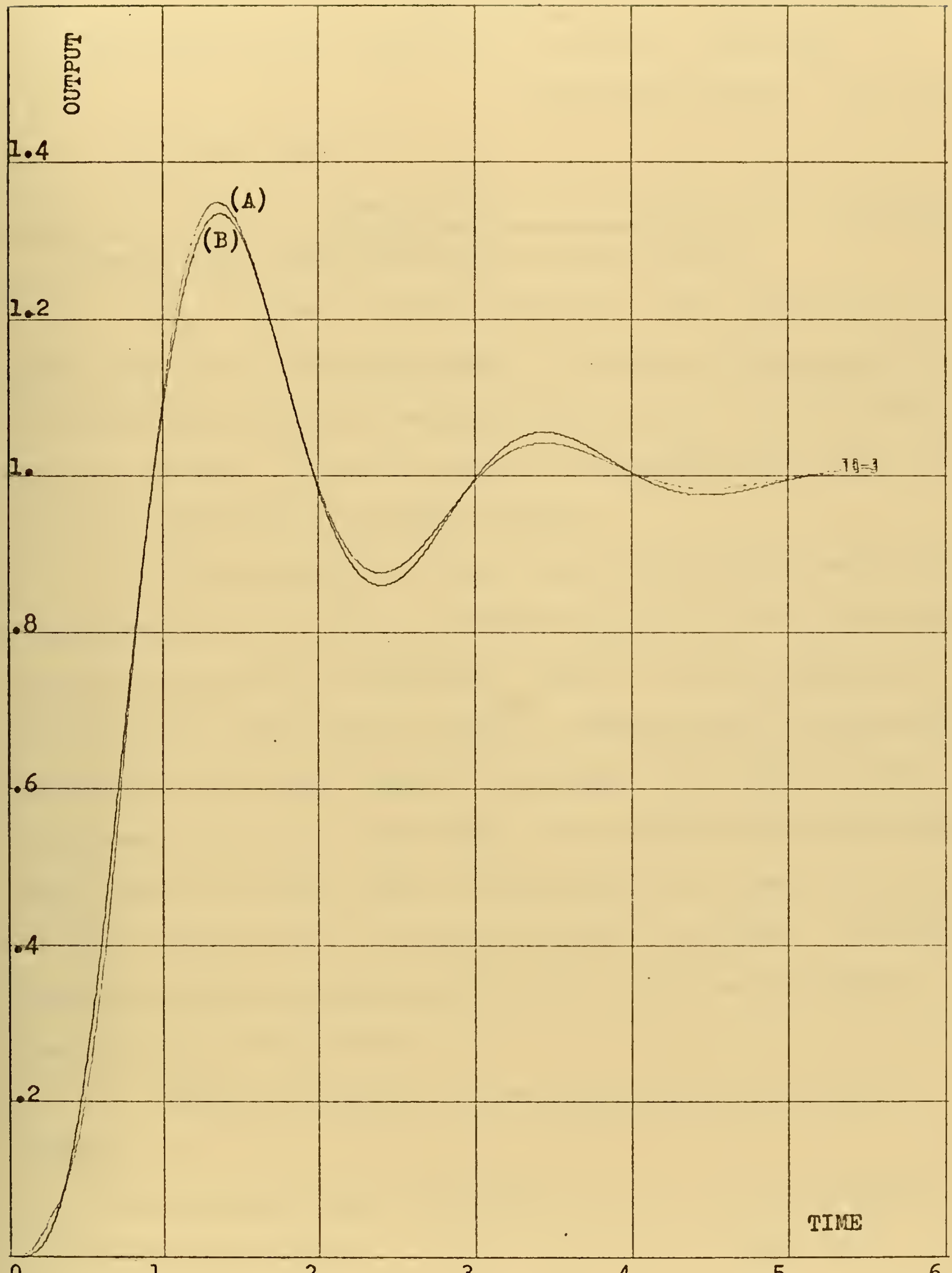

Figure III.36. EXAMPLE II. System's response (A) and the Six Poles and Three Zeros model's response (B) to a unit step input.

$$G(s)=\frac{384 \times 10^7}{s^7+432s^6+62670s^5+3615900s^4+75114000s^3+553920000s^2 \ldots \ldots +1443200000s+384\times 10^7} \quad \text{(III.17)}$$

or in factorized form

$$G(s)=\frac{384 \times 10^7}{(s^2+2s+10)(s+10)(s+20)(s+80)(s+120)(s+200)} \quad \text{(III.18)}$$

The response of this system to a unit step was computed from 0 to 8.4 seconds and 700 samples of this response (at intervals of 0.012 seconds) were used for the determination of optimum l^{th}-order models by minimizing the objective function specified in Chapter II.B.3.

It was decided to obtain all the low-order models of the seventh-order system described by Eq. (III.17) following the same technique as Examples I and II. First obtaining a second-order model with no zeros. Having found the optimum parameters using the computer programme, the order is then increased by one and a new model is selected and determined. When the sixth-order model is determined the number of zeros is also increased by one and new models are found. In this way the search is completed when the six poles and three zeros model is determined.

The main features of the step response are given below:

Time to reach first overshoot ....1.18 seconds

Maximum overshoot 30%

Steady-state value 1

Initial slope 0

Response at maximum overshoot 1.33

Rise time ....0.7 seconds

2. <u>The Model Response</u>

The desired model response was represented by two dominant complex conjugated roots and a number of real poles depending on the model's order.

From the curves of the step-response of a second order system with no finite zeros and the above features, starting parameters for the second order-model were selected as:

$$\alpha_1 = \zeta = 0.3$$

$$\alpha_2 = \omega_n = 3.5$$

and $t_f = 8.4$ seconds. In the same way using the universal curves of the second-order linear system step response characteristic, reasonable bounds were chosen as:

$$\begin{aligned} &0.1 \leqslant \alpha_1 \leqslant 0.55 \\ &1.5 \leqslant \alpha_2 \leqslant 6. \end{aligned} \tag{III.19}$$

in order to set constraints on the free parameters (α_1, α_2) for the determination of the second-order model, which will be starting parameter values (Chapter II.C) of the increasing order models.

The state equations of the system are written directly from Eq. (III.17).

The error criterion defined by Eq. (III.18) is minimized by the computer subroutine.

3. The Second-Order Model

The state equations can be written as Eq. (III.4).

The Bode gain was kept constant with no steady-state error between both responses (system and model).

Constraints were set on the free parameters as indicated in inequalities (III.19).

The optimum parameter values and minimum error criterion are tabulated in Table III.11, and the Fig. III.37 shows a plot of the error criterion (J) value.

4. Models

As was done in Examples I and II, all the different order models ranging from the two poles and no zero to the six poles and three zero were determined.

Also as a basis of comparison, the seven poles no zeros model was determined; the optimum parameter values for this model are very close to the real parameter values of the seventh-order test system, and the value of the performance index (J) is the smallest of all (really it would be zero but for the limitations on computer accuracy).

In Tables III.11 to III.14 are shown the different sets of free parameter values for each of the optimum specific-order models and the corresponding error criterion values.

Plots showing the variation of J versus the number of poles of the model are given in Figs. III.37 and III.38.

In order to get a basis of comparison between models and resultant values of the cost function (J) for all of

7	# Poles	2	3	4	5	6	7
.316	ζ	.37	.29	.309	.319	.321	.319
3.162	w_N	2.684	3.135	3.156	3.158	3.155	3.159
10	p_3		5.32	7.91	13.83	14.75	11.12
20	p_4			18.66	14.72	17.	19.34
80	p_5				36.73	33.34	78.81
120	p_6					107.75	119.1
200	p_7						181.2
Test System	Jx 10^4	161.	4.1	0.45	.31	.404	.3

TABLE III.11. EXAMPLE III. Optimum models with no zeros.

7	# Poles	2	3	4	5	6	7
.316	ζ	.371	.296	.31	.312	.317	.319
3.162	w_N	2.673	3.09	3.163	3.164	3.162	3.159
10	p_3		6.18	10.4	9.37	12.6	12.43
20	p_4			11.18	14.22	12.94	16.37
80	p_5				77.7	63.13	66.8
120	p_6					128.4	90.
200	p_7						160.7
----	z_1	249.9	219.3	79.1	57.73	92.19	164.8
Test System	Jx 10^4	168.	4.5	0.52	0.41	0.27	0.31

TABLE III.12. EXAMPLE III. Optimum models with one zero.

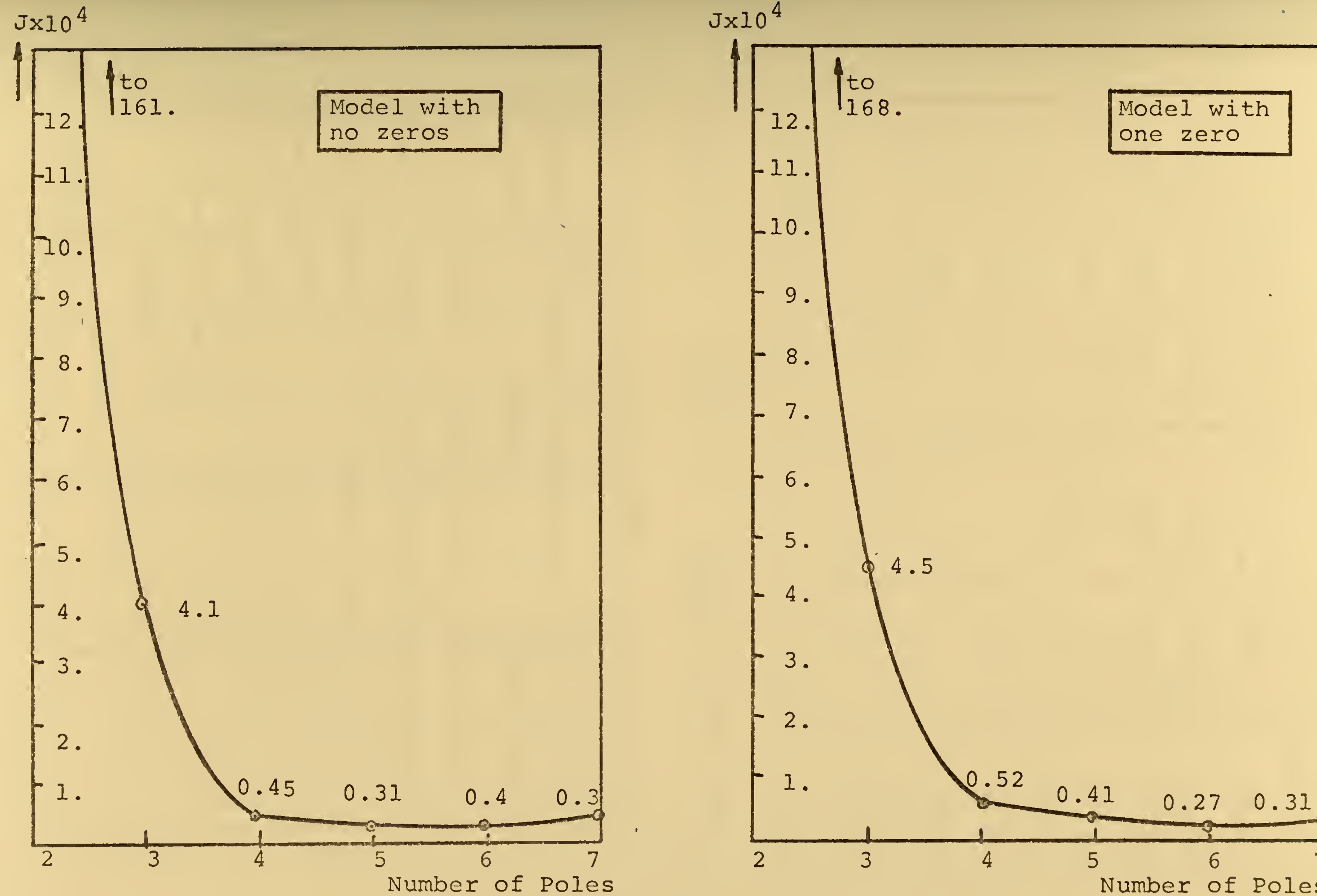

Figure III.37.

7	# Poles	3	4	5	6
.316	ζ	.285	.304	.309	.314
3.162	w_N	3.16	3.161	3.165	3.163
10	p_3	4.95	8.54	8.84	9.42
20	p_4		10.34	12.75	22.2
80	p_5			71.86	31.16
120	p_6				37.9
200	z_1	99.7	46.55	47.75	36.5
----	z_2	238.3	65.33	103.1	89.76
Test System ↑	$J \times 10^4$	5.5	1.3	.65	.29

TABLE III.13. EXAMPLE III. Optimum models with two zeros.

7	# Poles	4		
.316	ζ	.306		
3.162	w_N	3.168		
10	p_3	8.55		
20	p_4	11.41		
80	p_5			
120	p_6			
200	z_1	94.5		
---	z_2	99.5		
---	z_3	140.47		
Test System ↑	$J \times 10^4$	1.		

TABLE III.14. EXAMPLE III. Optimum model with three zeros.

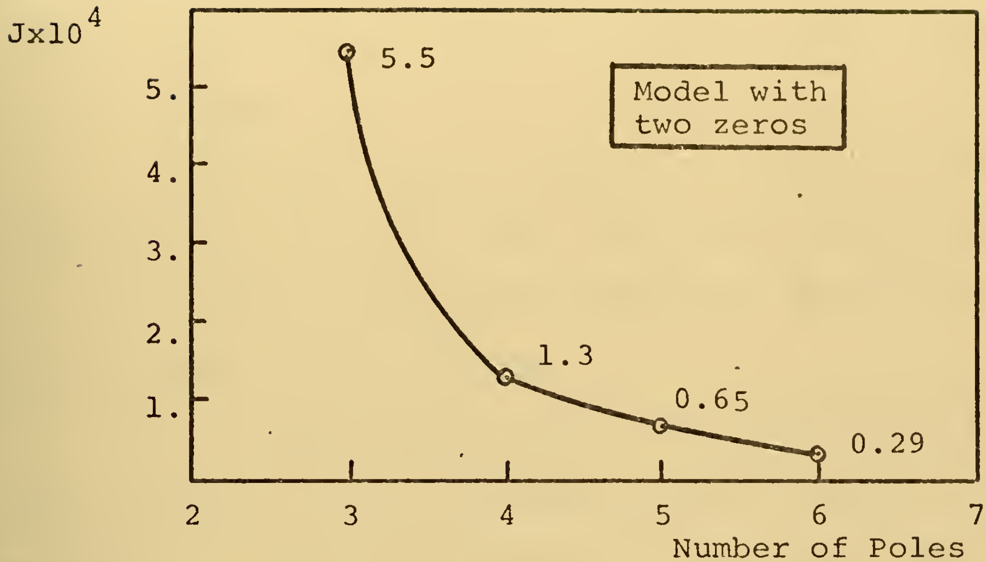

Figure III.38.

# Z \ # P	2	3	4	5	6
0	161.	4.1	0.45	0.31	0.404
1	168.	4.5	0.52	0.41	0.27
2		5.5	1.3	0.65	0.29
3			1.005	--	--

TABLE III.15. Error criterion ($Jx10^4$) for models in Example III.

them are tabulated in Table III.15. Figures III.39 to III.49 show the transient responses of both the system and the reduced model for different pole and zero values.

5. Remarks

Some facts can be pointed out looking at Tables III.11 to III.15 and at Figs. III.37 and III.38, such as:

a. The second-order models (no zero and one zero) do not appear to be suitable.

b. A model which gives a minimum value of the error criteria is the six poles one zero model, although any model in the range from four to six poles (and even the third-order model) give small values of the error criteria.

c. As the order of the model is increased the locations of optimum poles values approaches the real values. Since the error criterion remains almost the same in the range from four to six poles, the poles located far away contribute very little to the general transient response (performance of the system).

d. If it is desired to represent the system by a specific order model, the tables indicate the cost paid for the simplicity. Similar tables can be constructed for any specific problem.

E. EXAMPLE IV

1. General

It was decided to select the model of a real physical system in order to continue the investigation of the reduced

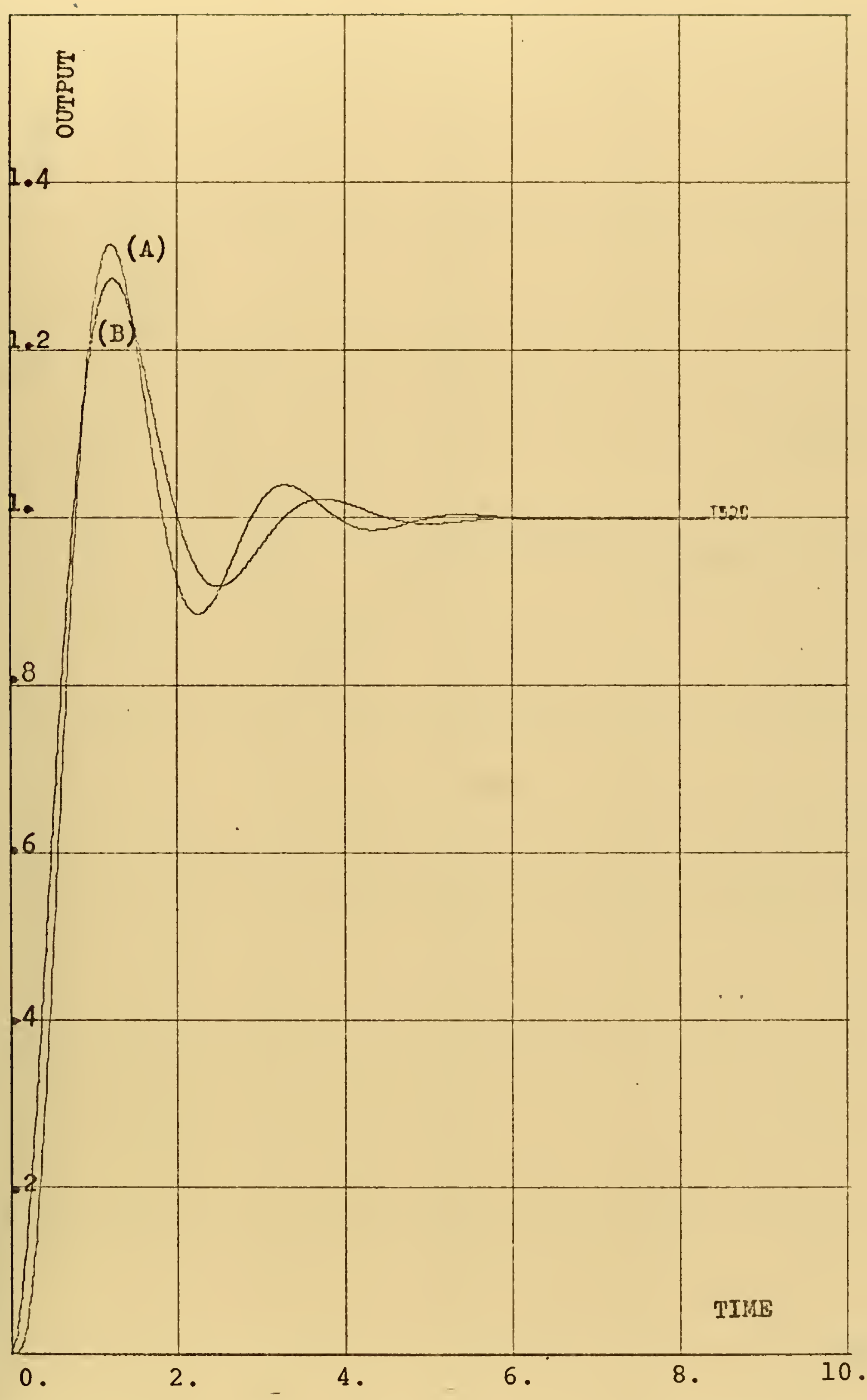

Figure III.39. EXAMPLE III. System's response (A) and the Two Poles no Zero model's response (B) to a unit step input.

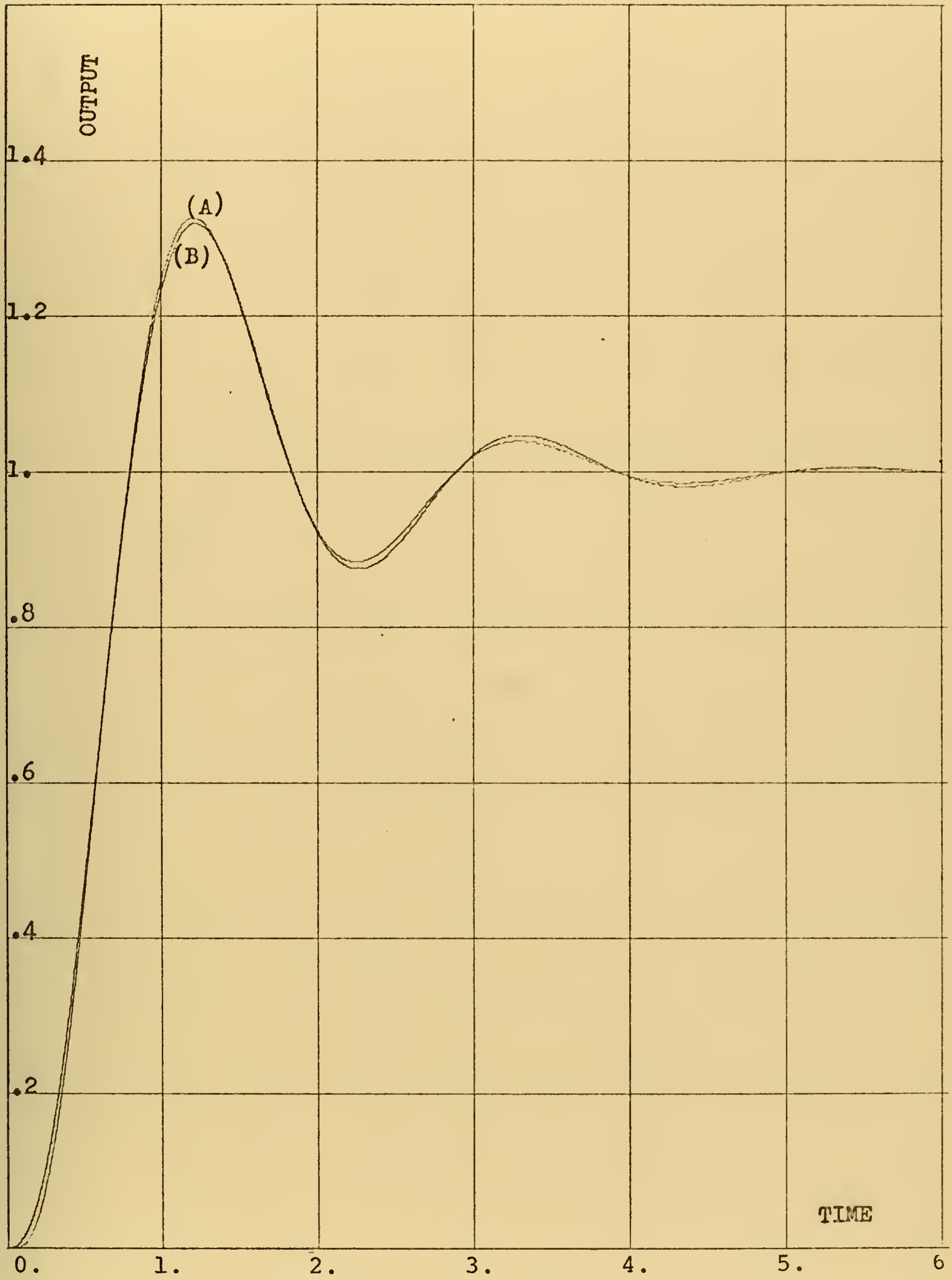

Figure III.40. EXAMPLE III. System's response (A) and the Three Poles no Zero model's response (B) to a unit step input.

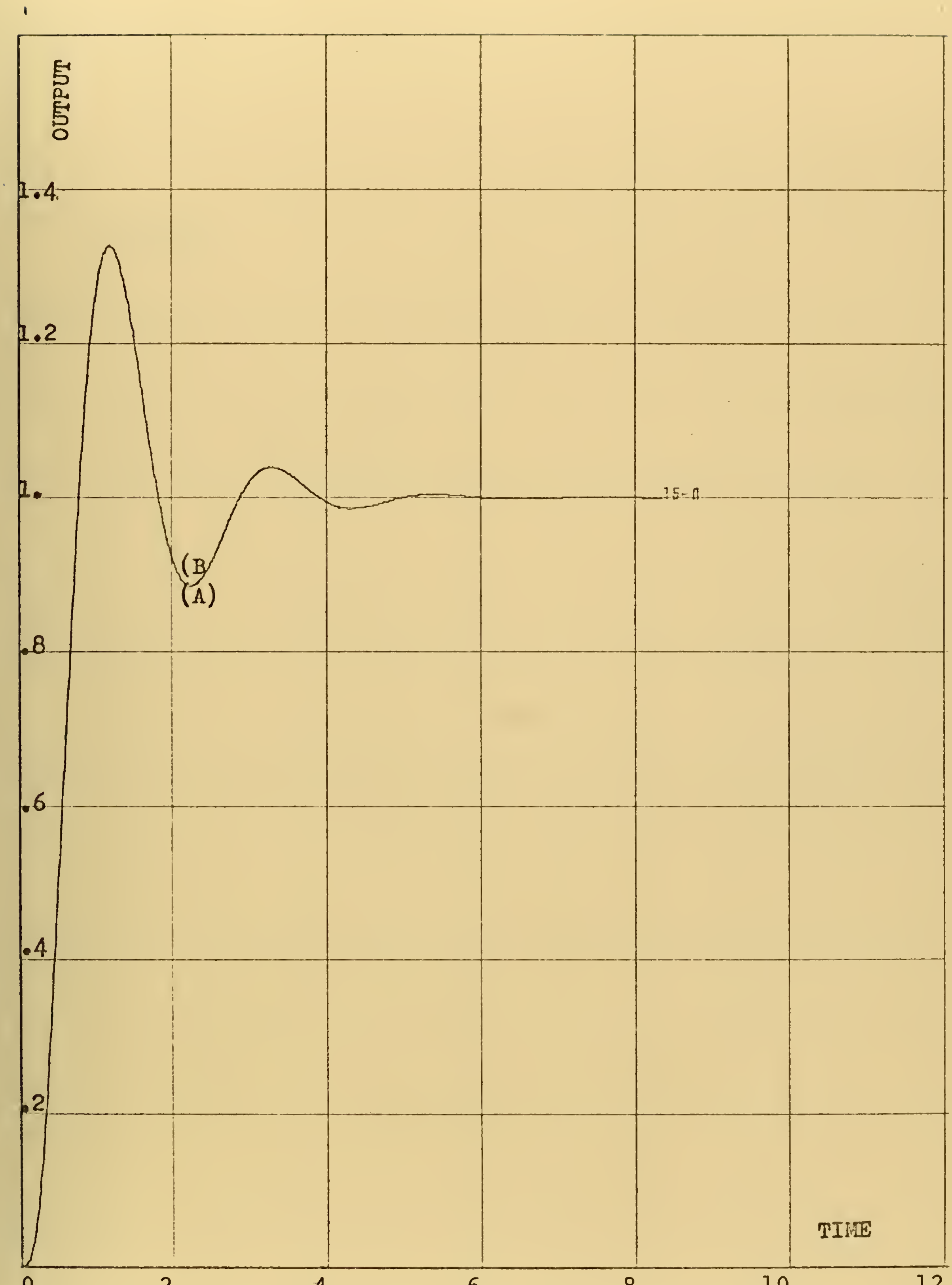

Figure III.41. EXAMPLE III. System's response (A) and the Five Poles no Zero model's response (B) to a unit step input.

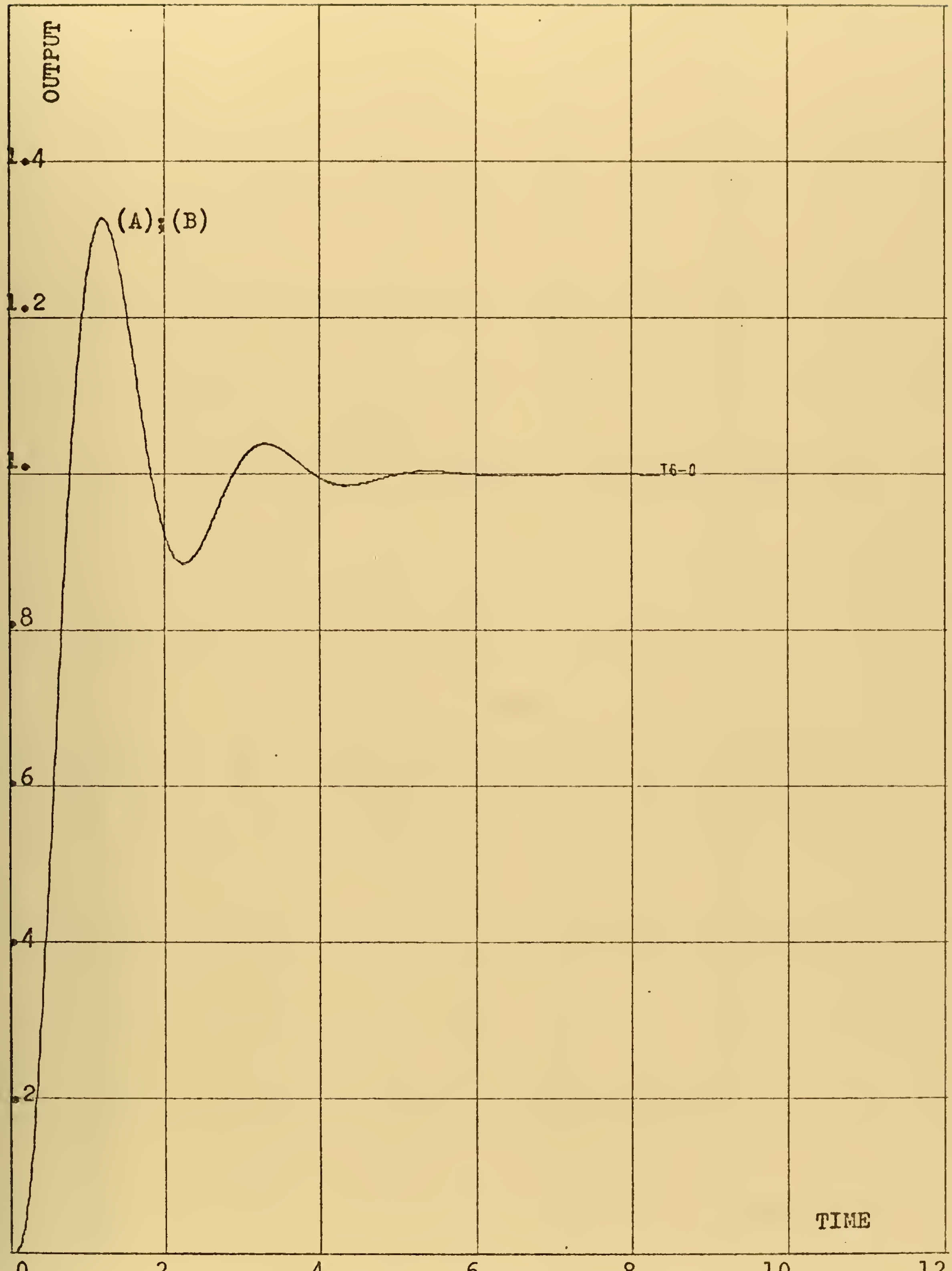

Figure III.42. EXAMPLE III. System's response (A) and the Six Poles no Zero model's response (B) to a unit step input.

Figure III.43. EXAMPLE III. System's response (A) and the Three Poles and One Zero model's response (B) to a unit step input.

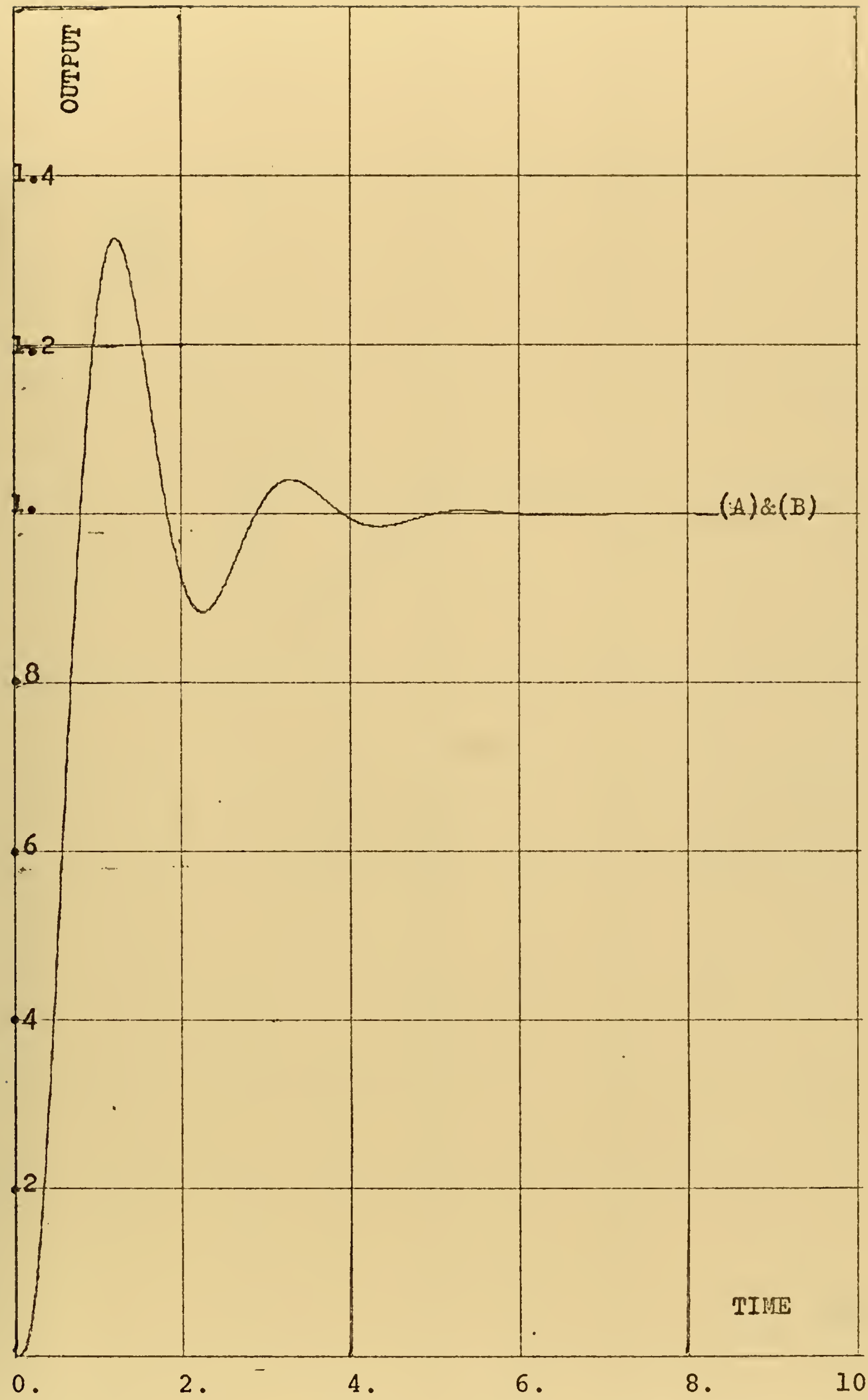

Figure III.44. EXAMPLE III. System's response (A) and the Five Poles and One Zero model's response (B) to a unit step input.

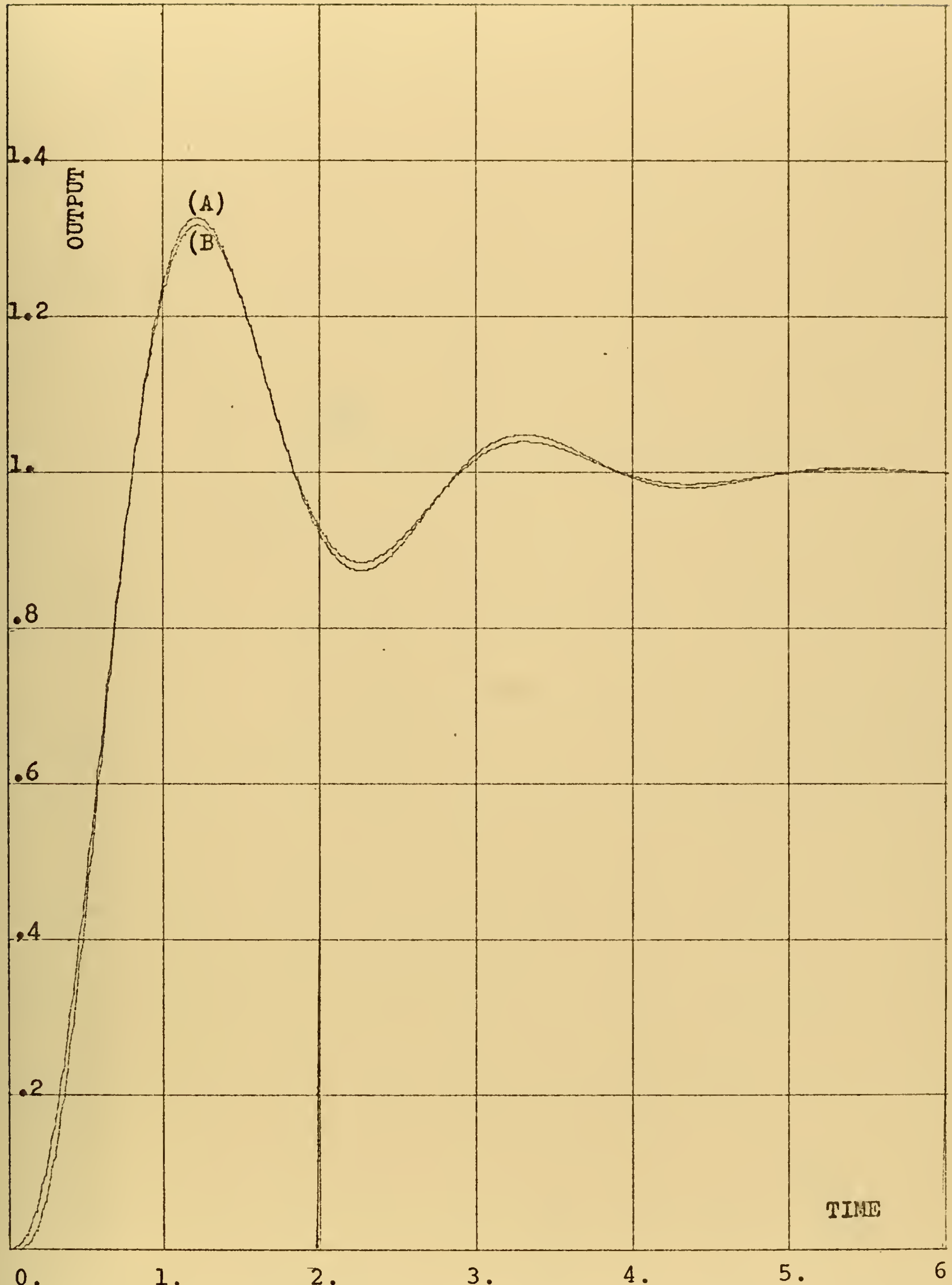

Figure III.45. EXAMPLE III. System's response (A) and the Three Poles and Two Zeros model's response (B) to a unit step input.

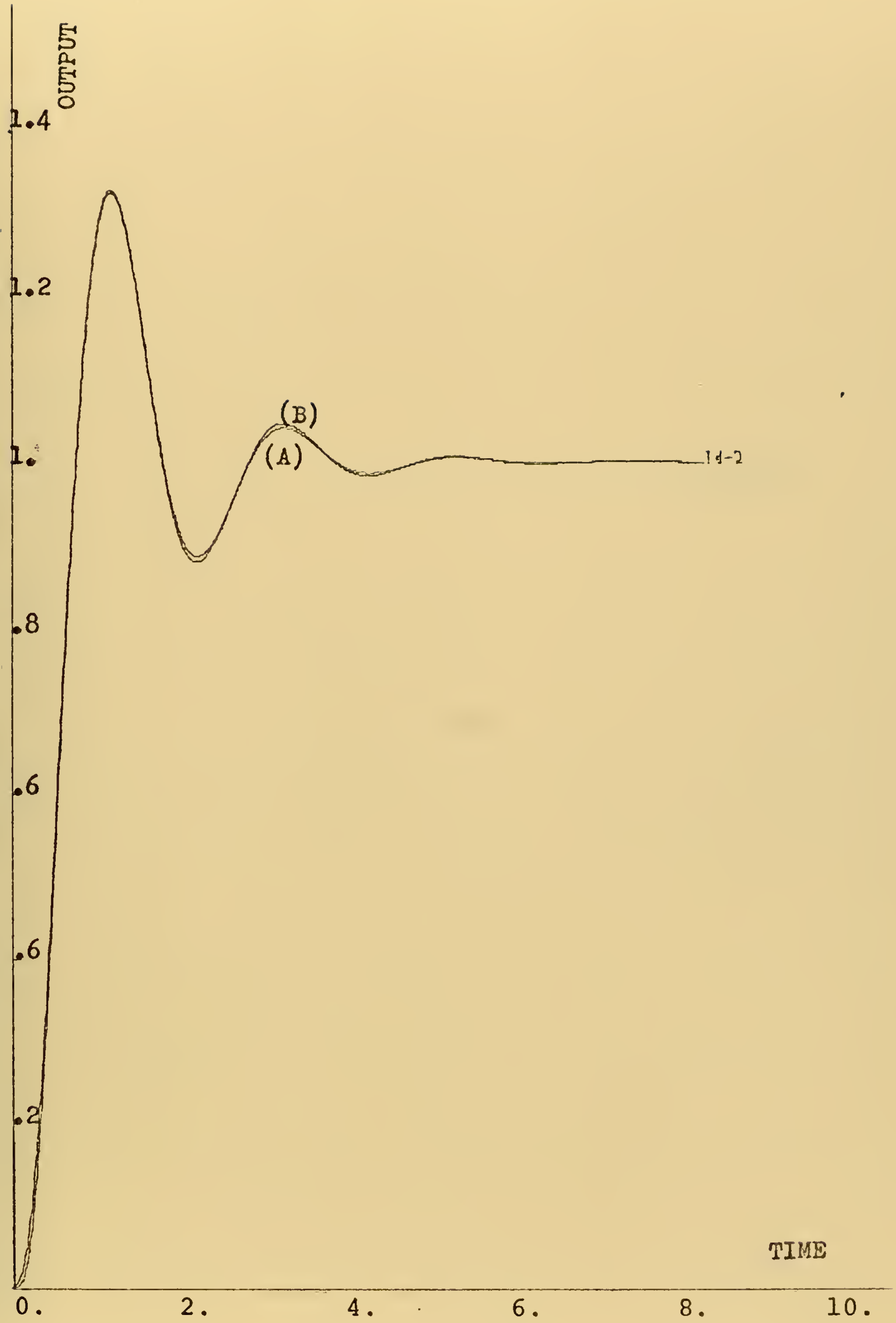

Figure III.46. EXAMPLE III. System's response (A) and the Four Poles and Two Zeros model's response (B) to a unit step input.

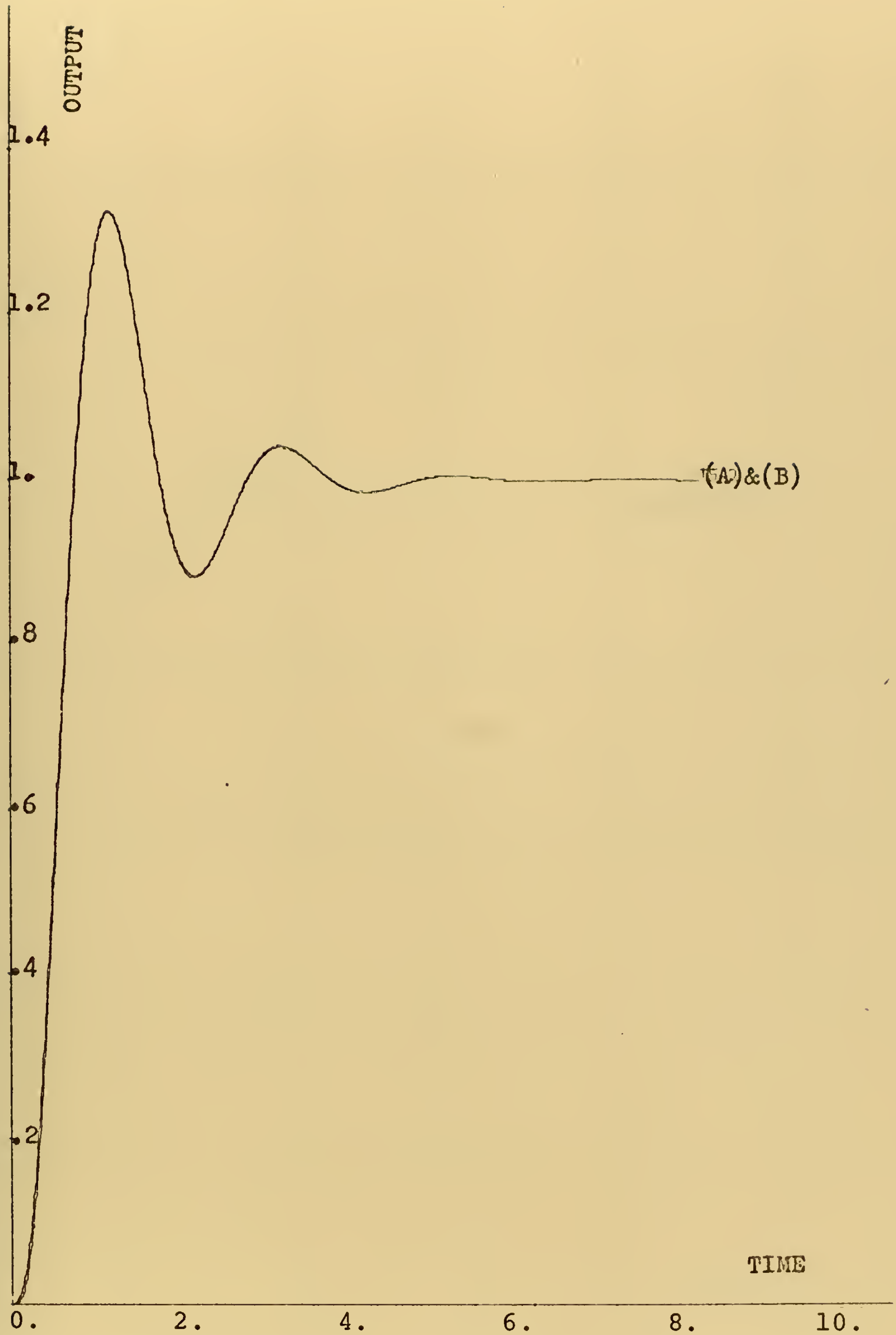

Figure III.47. EXAMPLE III. System's response (A) and the Five Poles and Two Zeros model's response (B) to a unit step input.

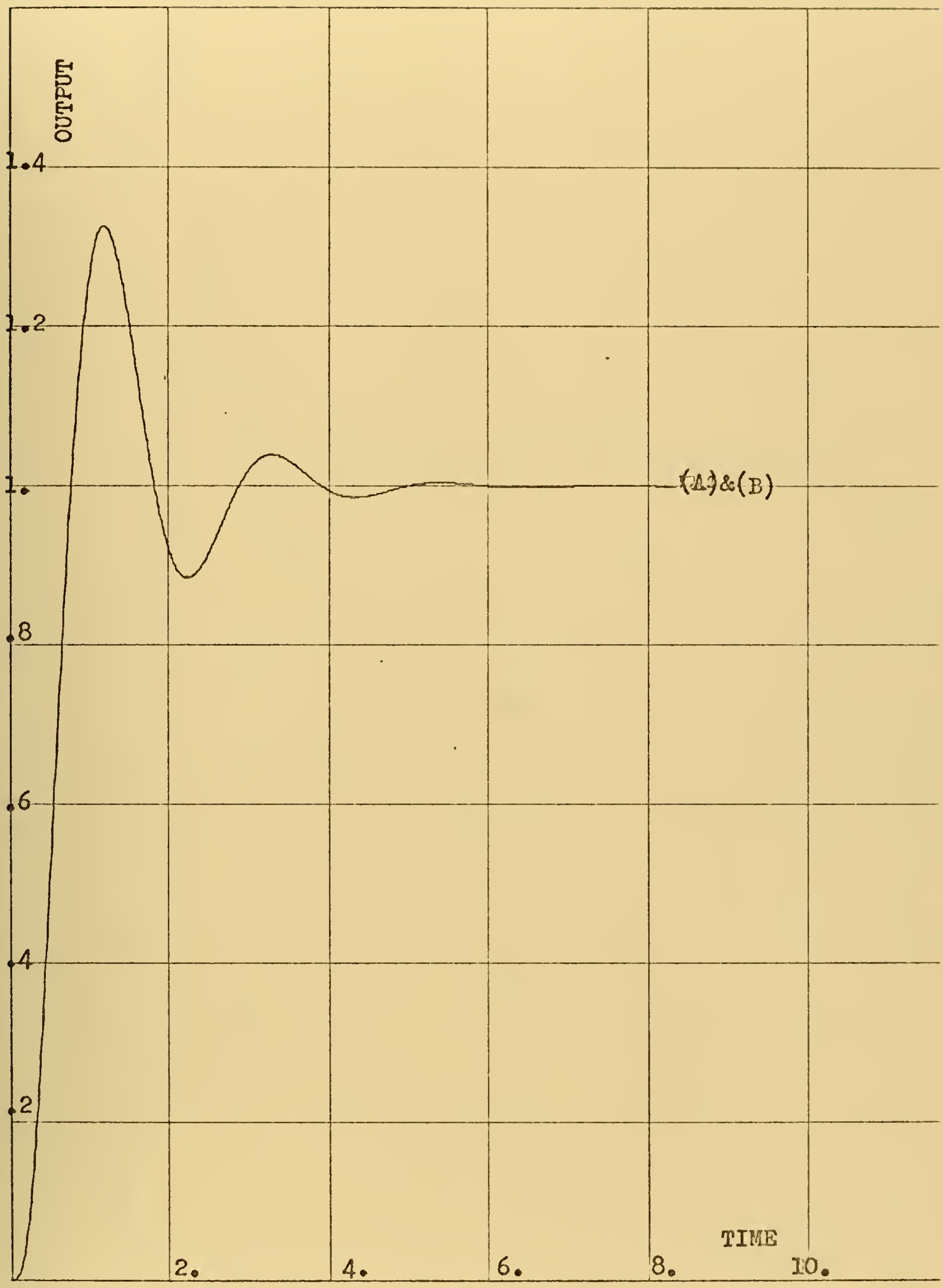

Figure III.48. EXAMPLE III. System's response (A) and the Six Poles and Two Zeros model's response (B) to a unit step input.

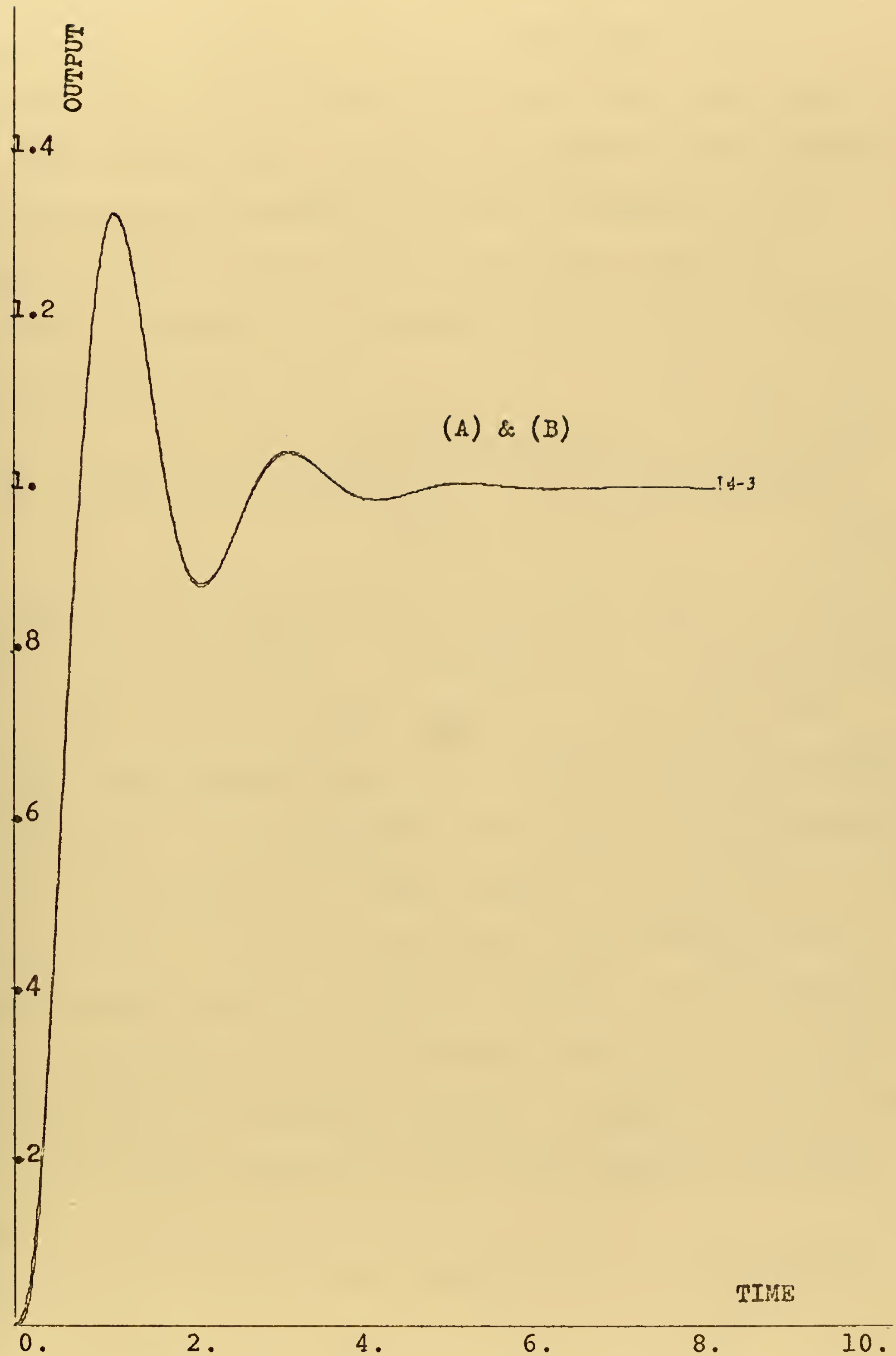

Figure III.49. EXAMPLE III. System's response (A) and the Four Poles and Three Zeros model's response (B) to a unit step input.

models for the given system. The system chosen was one representing the control system of the pitch rate of a supersonic aircraft, and was taken from Sinha and others [2]. Sinha selects the variable parameters in the system block diagram for reasonable pole-zero locations, consistent with design description and system stability. With these parameters, the transfer function of the test system is given by

$$\frac{Y(s)}{R(s)} = G(s) = \frac{375000(s+0.08333)}{s^7+83.64s^6+4092s^5+70342s^4+85370s^3+2814271s^2 \ldots \ldots 3310875s+281250} \qquad \text{(III.19)}$$

The roots of the characteristic polynomial are

-0.092 ; $-2.024 \pm j0.965$; $-7.675 \pm j13.445$; $-32.065 \pm j38.863$

The response of this system to a unit step was simulated and computed from 0 to 21 seconds and 700 samples (at intervals of 0.03 seconds) were used for the determination of all possible l^{th}-order models ($2<\ell<6$) with "m" zeros ($0<m<3$), by the same technique and minimizing the error criteria function defined in Chapter II.B as the integral of the squares of the output errors with no error in steady-state (constraint imposed) response to step input.

The main features of the step response are given below:

Time to reach first overshoot ......2.9 seconds

Maximum overshoot 8.6%

Steady state value 0.111

Response at maximum overshoot 0.12057

Rise time ....1 second

Because the step-response of the system has overshoot a pair of conjugate poles and (ℓ-2) real poles were specified for the different models.

The simple second-order model was determined as an approach for starting the study. The state equations of the model are

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) & ; \quad x_1(0)=0 \\ \dot{x}_2(t) &= 1.-\alpha_2^2 x_1(t) - 2\alpha_1\alpha_2 x_2(t) & ; \quad x_2(0)=0 \end{aligned} \qquad \text{(III.21)}$$

where the free parameters α_1 and α_2 represent the damping ratio and the natural frequency.

The output of the model is

$$y_r(t) = Kx_1(t) \qquad \text{(III.22)}$$

From Eq. (III.19) the state equations of the system can be written

$$\underline{\dot{x}}(t) = \begin{bmatrix} 0 & 1 & 0 & \cdots\cdots\cdots & 0 \\ 0 & 0 & 1 & \cdots\cdots\cdots & 0 \\ \cdot & \cdot & \cdot & & \\ \cdot & \cdot & \cdot & & \\ -a_1 & -a_2 & -a_3 & \cdots\cdots\cdots & -a_7 \end{bmatrix} \cdot \underline{x}(t) + \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ 1 \end{bmatrix} \cdot r(t)$$

$$\underline{y}(t) = K[0.0833 \quad 1 \quad 0 \quad . \quad . \quad 0] \cdot \underline{x}(t) \qquad \text{(III.23)}$$

From the step response of second-order linear system with no zeros and taking into account the features given in Eq. (III.20), good starting parameter values were selected as:

$$\begin{aligned} \alpha_1 &= \zeta = 0.7 \\ \alpha_2 &= w_N = 2.5 \\ t_f &= 21 \text{ seconds} \end{aligned} \qquad \text{(III.24)}$$

A performance index which is a function of the deviation between the output of both system and model, is defined as in Chapter II.B. Minimization of this cost function with respect to α_1 and α_2 yields the "best" second-order model.

Equations (III.21) and (III.23) are integrated with respect to time to calculate the value of the error criterion at each step of the minimization process. The integration step size is chosen as 0.03 seconds, the same sample interval of the system. The same criterion on the Bode gain, steady-state error and starting parameter boundaries with flexibility to change them yields the results tabulated in Table III.16. The transfer function of the second-order model becomes:

$$G_r(s)=\frac{0.36886}{s^2+2.3322s+3.3197} \qquad \text{(III.25)}$$

and the pole locations: $-1.1661 \pm j1.424$.

After the above second-order model was determined the various models of third, fourth, fifth and sixth orders were obtained. The errors produced by the various models are shown in Table III.20 and are plotted in Fig. III.50. For further comparison between the reduced models a complete list of the optimum parameter values are tabulated in Tables III.16 to III.19.

Step responses of some models and the actual system are shown in Figs. III.51 to III.58 on identical time scales.

Poles / Zeros	1	2	3	4	5	6
0		4.7	4.6	4.6	4.9	5.1
1		4.6	2.7	2.5	3.	2.7
2			4.7	3.1	2.9	2.48
3				2.7	2.59	2.1

TABLE III.20. Error criterion ($Jx10^4$) for various models in Example IV.

It will be seen that most of the step responses of the determined models are not a good approximation to the true response of the tested system. Larger overshoot and smaller settling time can be observed and a singular fact can be pointed out from the step response figures: the decaying part of the transient response was not achieved by the determined models having only a pair of complex conjugate poles and the others (ℓ-2) poles restricted to be real. It is clear that the actual system has a small real pole which has not been identified.

Looking at Tables III.16 to III.19 is noticed that the third pole (first real pole) of the different models is always greater than 5.2, no optimum value less than this value is obtained. The selected system has a real pole very close to the origin, $p_1 = 0.092$, and also has one zero, $z_1 = 0.0833$. It was thought that the effect on the transient response of this zero and pole would be minimum due to the fact that they are close together and the zero might cancel the residue at the pole so that the contribution to the step-response would be negligible. Actually the pole close

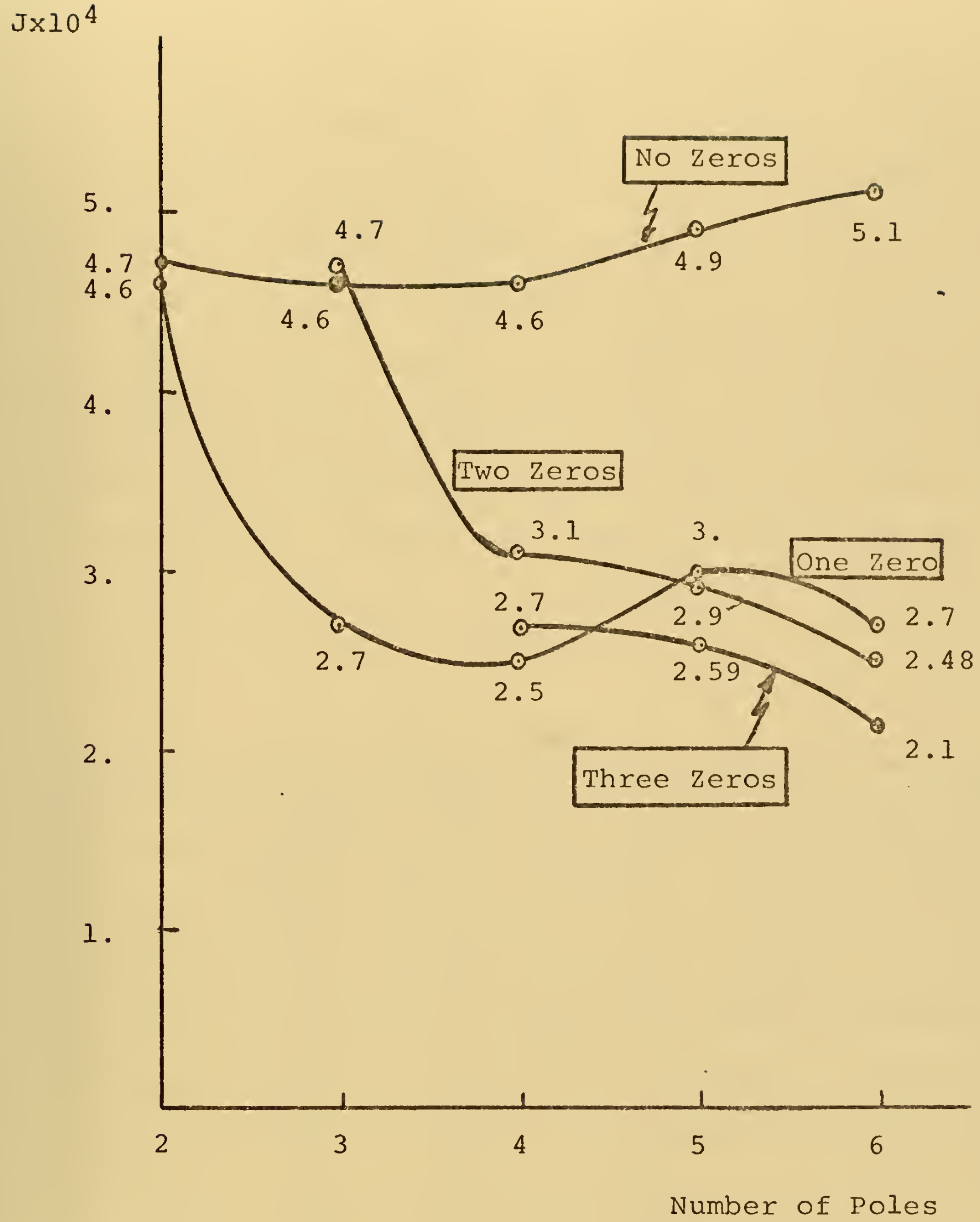

Figure III.50. Error criterion versus number of poles for Example IV.

# p	2	3	4	5	6
ζ	.64	.648	.664	.669	.674
w_N	1.822	1.854	1.997	2.035	2.068
p_3		80.75	25.	24.93	29.81
p_4			49.65	49.93	44.88
p_5				89.83	69.86
p_6					92.64
$J\times 10^4$	4.7	4.6	4.6	4.9	5.1

TABLE III.16. EXAMPLE IV. Optimum models with no zeros.

# p	2	3	4	5	6
ζ	.609	.868	.98	.894	.858
w_N	1.486	.764	.7	.753	.8
p_3		5.2	8.34	6.38	12.61
p_4			9.72	44.02	22.56
p_5				44.4	25.89
p_6					46.82
z_1	5.77	.477	.371	.454	.52
$J\times 10^4$	4.6	2.7	2.5	3.	2.7

TABLE III.17. EXAMPLE IV. Optimum models with one zero.

# p	3	4	5	6
ζ	.624	.885	.719	.89
w_N	1.639	1.195	.654	.759
p_3	14.763	.849	3.043	10.91
p_4		18.43	5.677	11.24
p_5			7.031	16.18
p_6				69.86
z_1	9.44	3.687	.746	.454
z_2	48.75	5.0	.876	23.8
$J \times 10^4$	4.7	3.1	2.9	2.48

TABLE III.18. EXAMPLE IV. Optimum models with two zeros.

# p	4	5	6
ζ	.894	.895	.98
w_N	.755	.731	.694
p_3	6.388	1.02	7.55
p_4	32.9	5.21	7.15
p_5		15.16	10.19
p_6			29.3
z_1	.454	.442	.363
z_2	.755	.953	7.95
z_3	.894	20.1	18.72
$J \times 10^4$	2.7	2.59	2.1

TABLE III.19. EXAMPLE IV. Optimum models with three zeros.

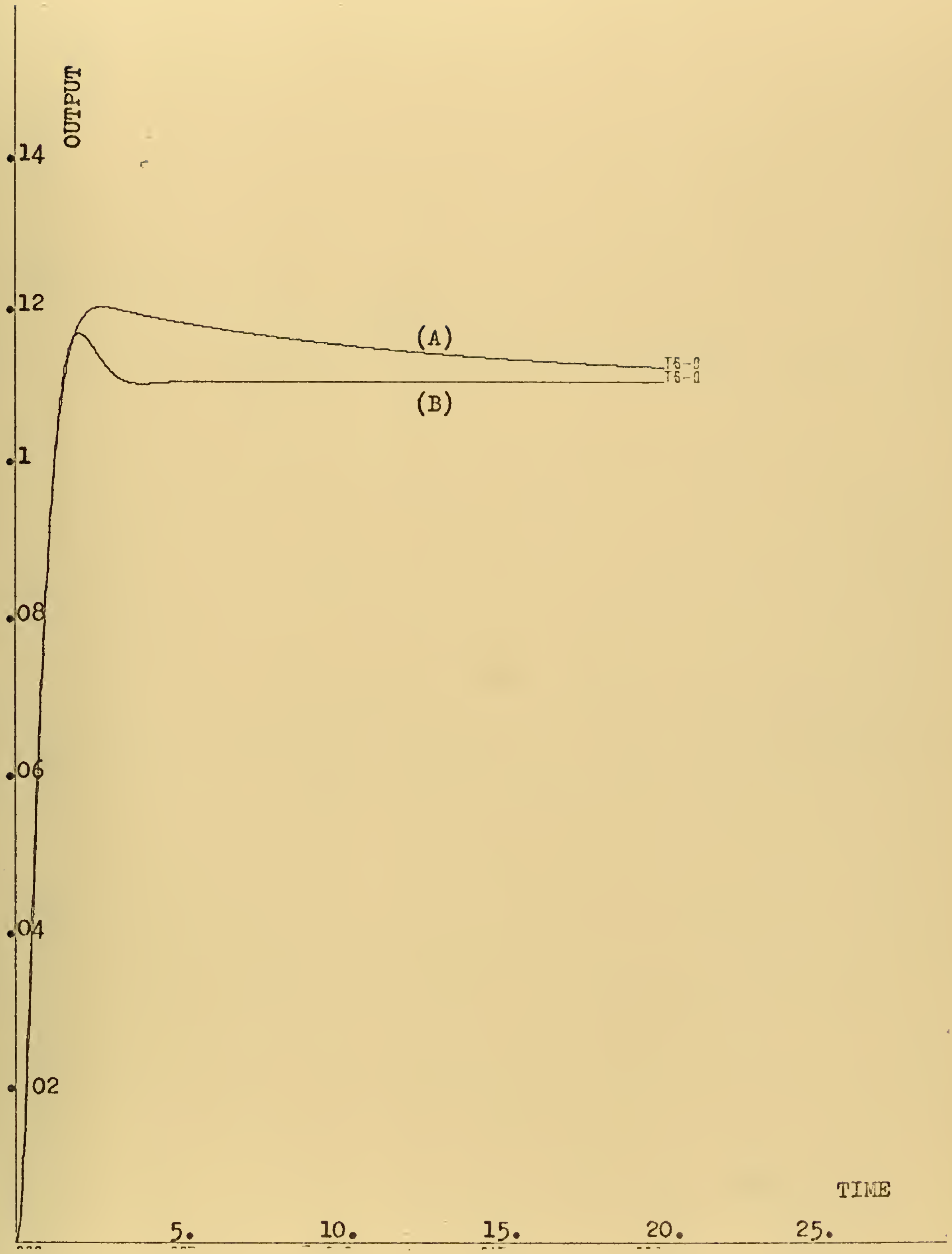

Figure III.51. EXAMPLE IV. System's response (A) and the Two Poles no Zero model's response (B) to a unit step input.

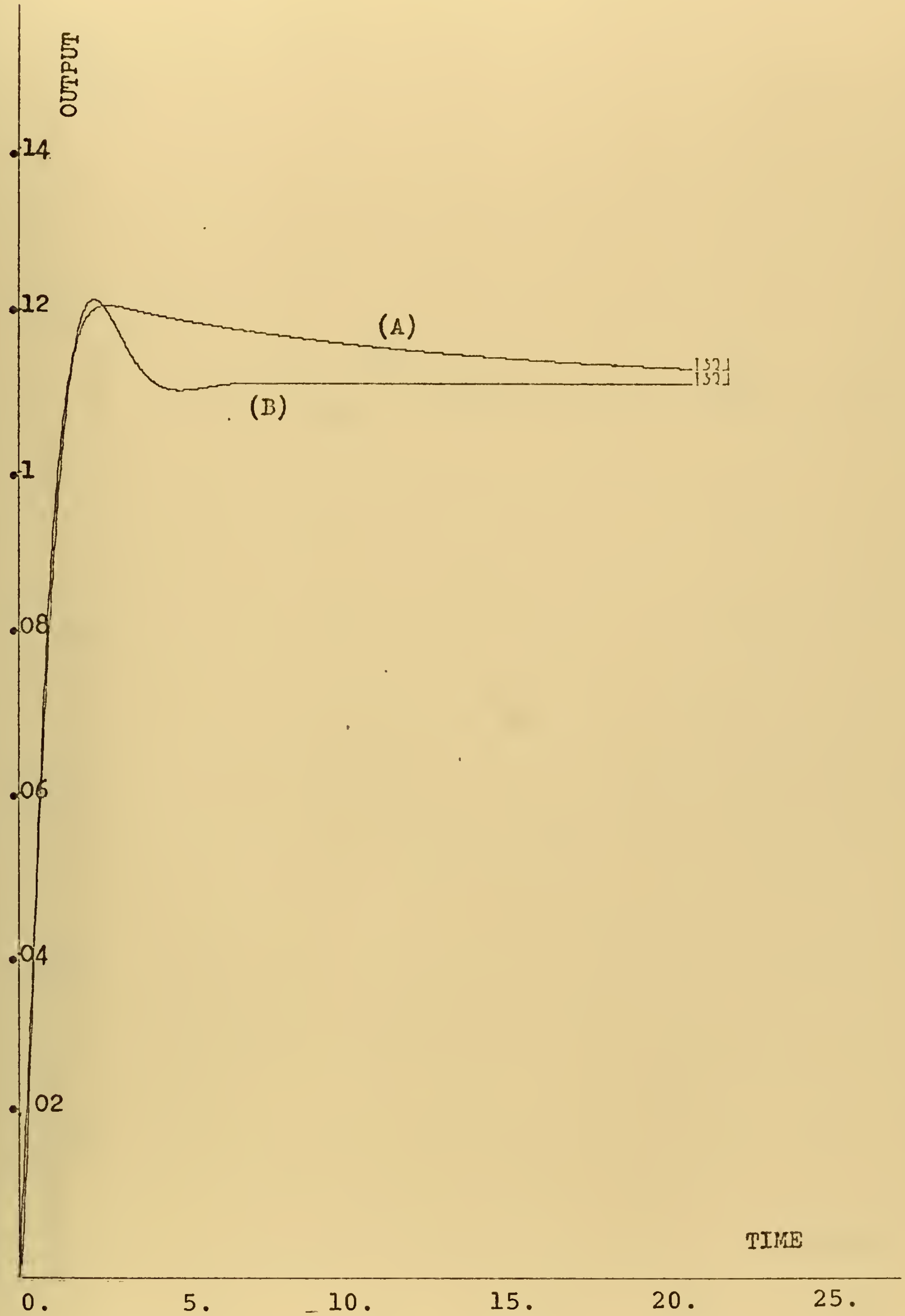

Figure III.52. EXAMPLE IV. System's response (A) and the Two Poles and One Zero model's response (B) to a unit step input.

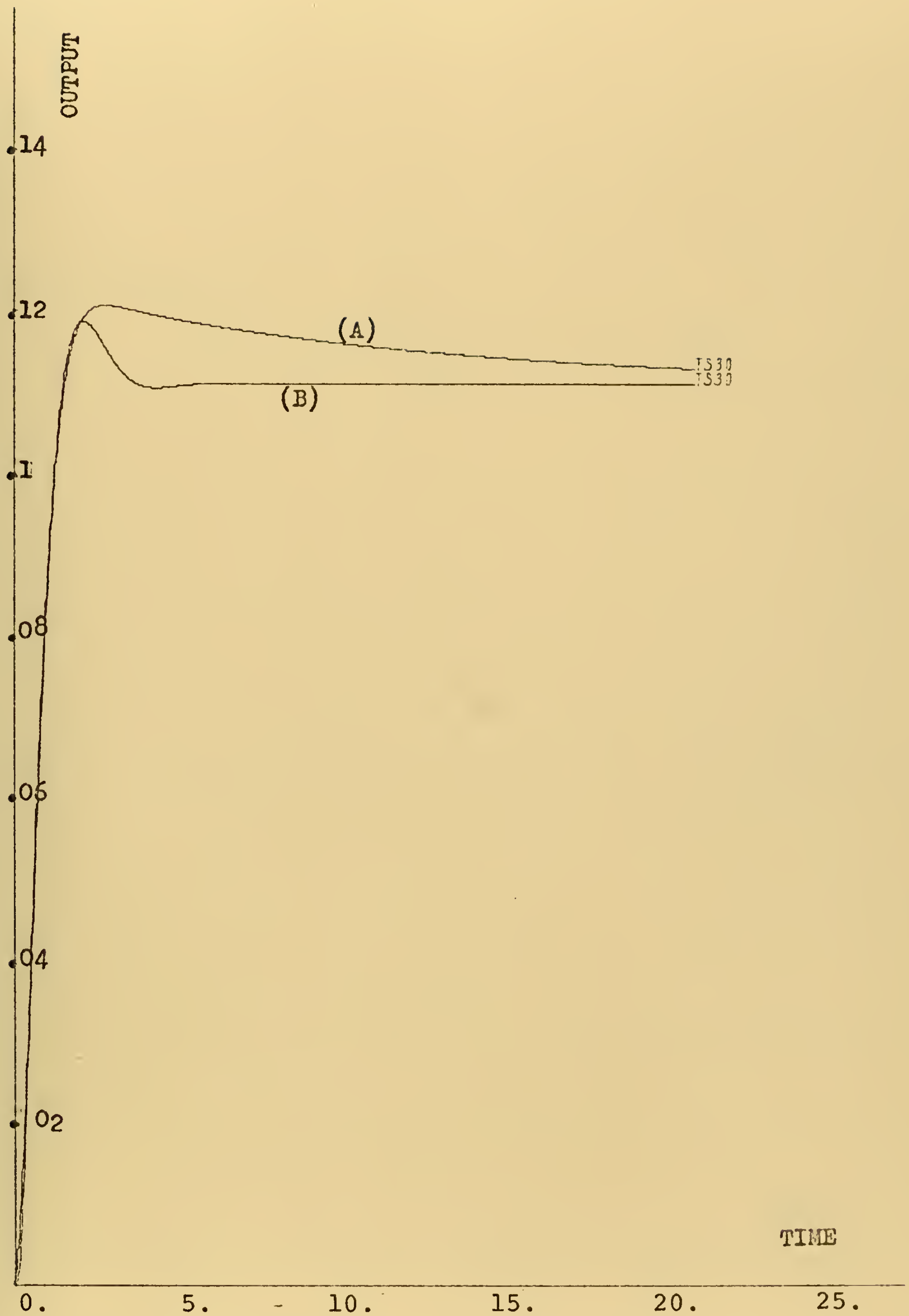

Figure III.53. EXAMPLE IV. System's response (A) and the Three Poles no Zero model's response (B) to a unit step input.

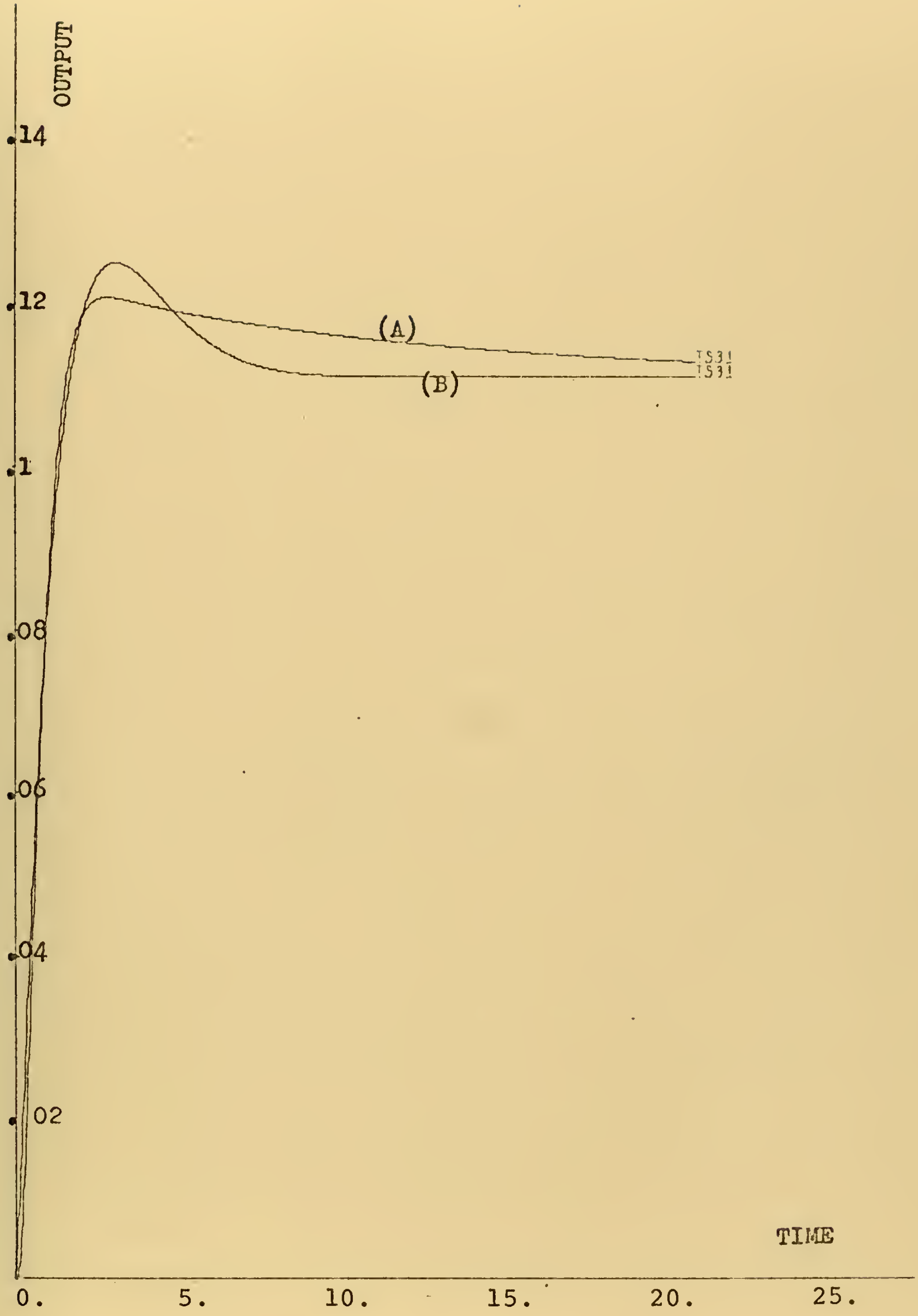

Figure III.54. EXAMPLE IV. System's response (A) and the Three Poles and One Zero model's response (B) to a unit step input.

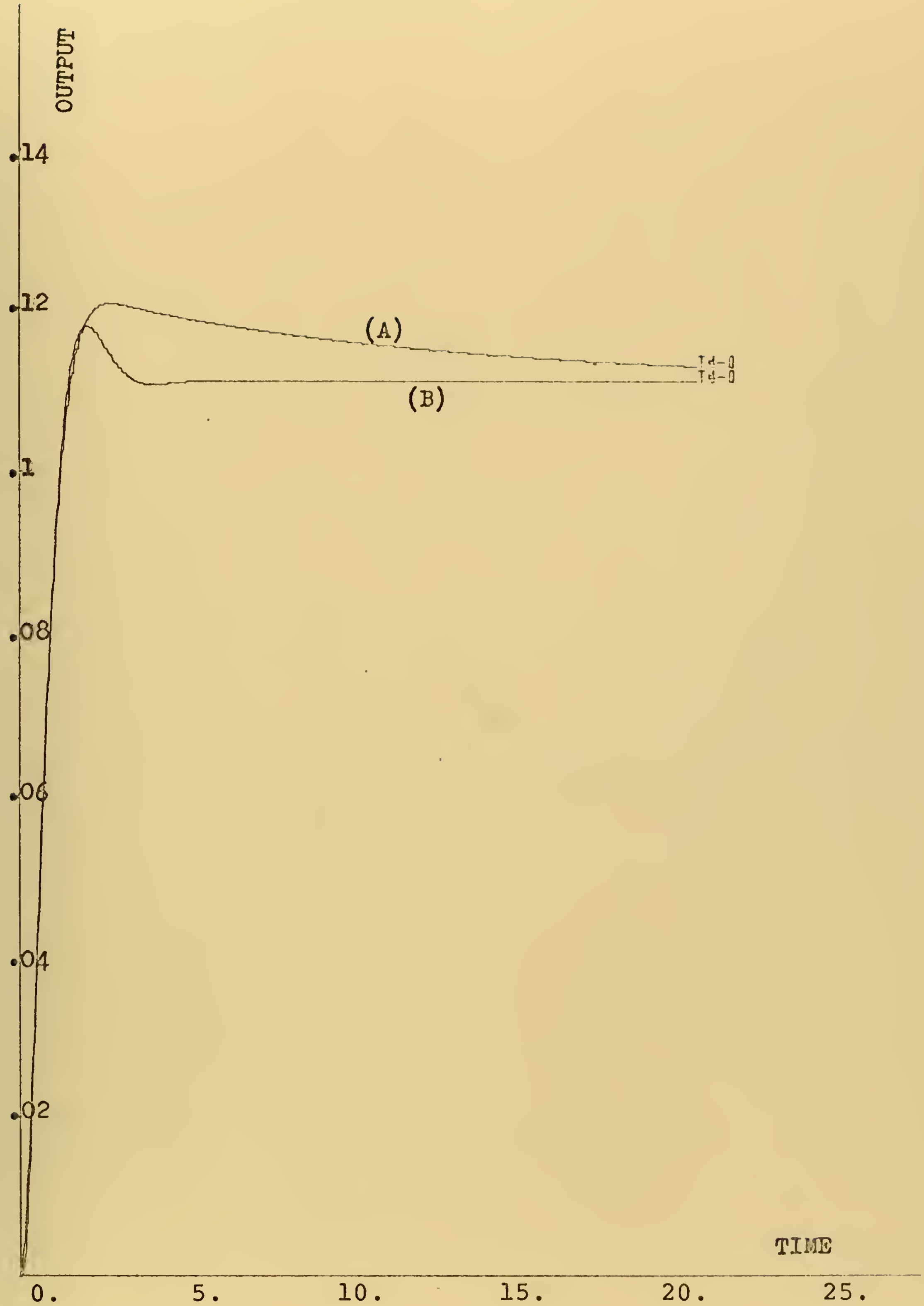

Figure III.55. EXAMPLE IV. System's response (A) and the Four Poles and no Zero model's response (B) to a unit step input.

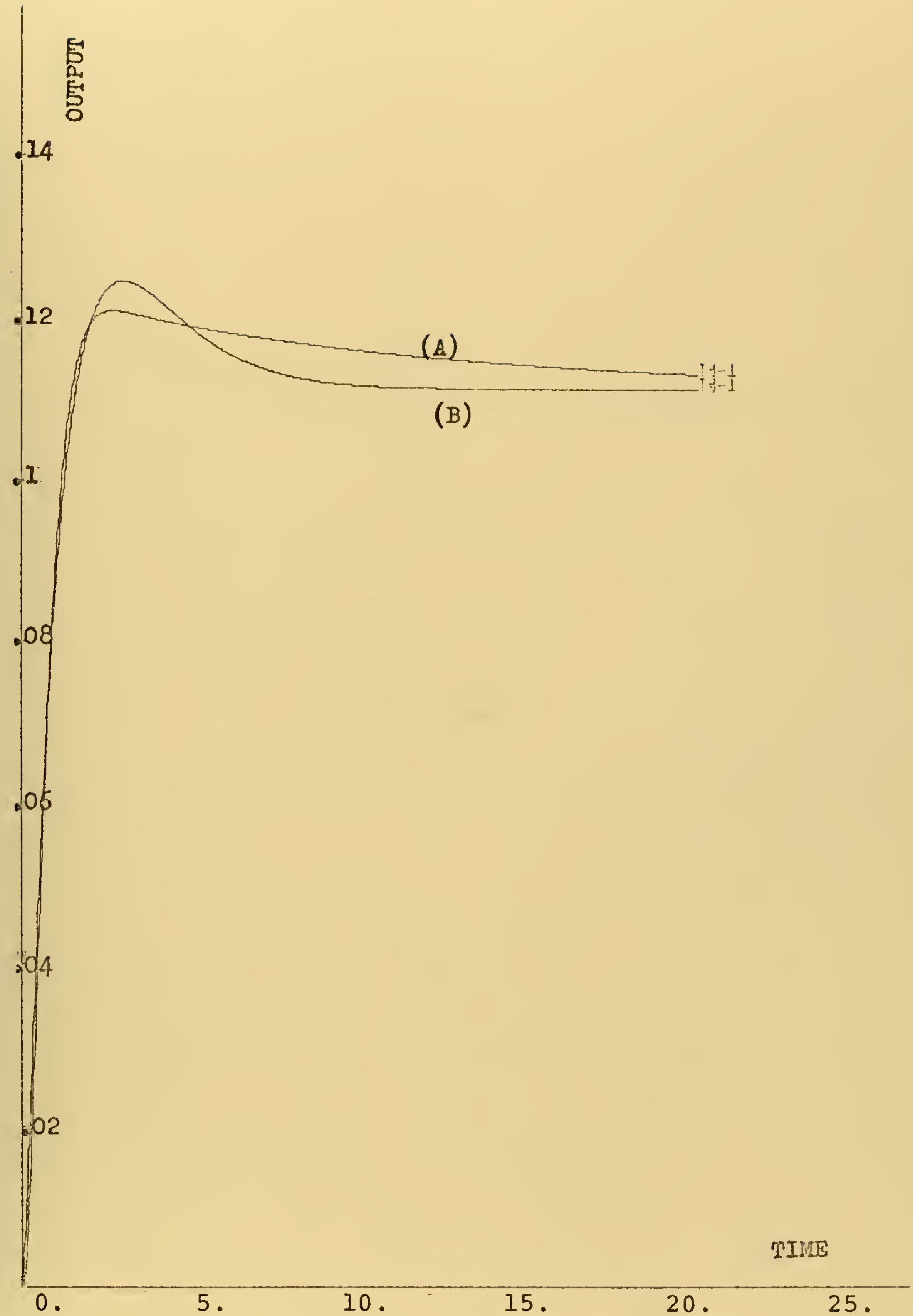

Figure III.56. EXAMPLE IV. System's response (A) and the Four Poles and One Zero model's response (B) to a unit step input.

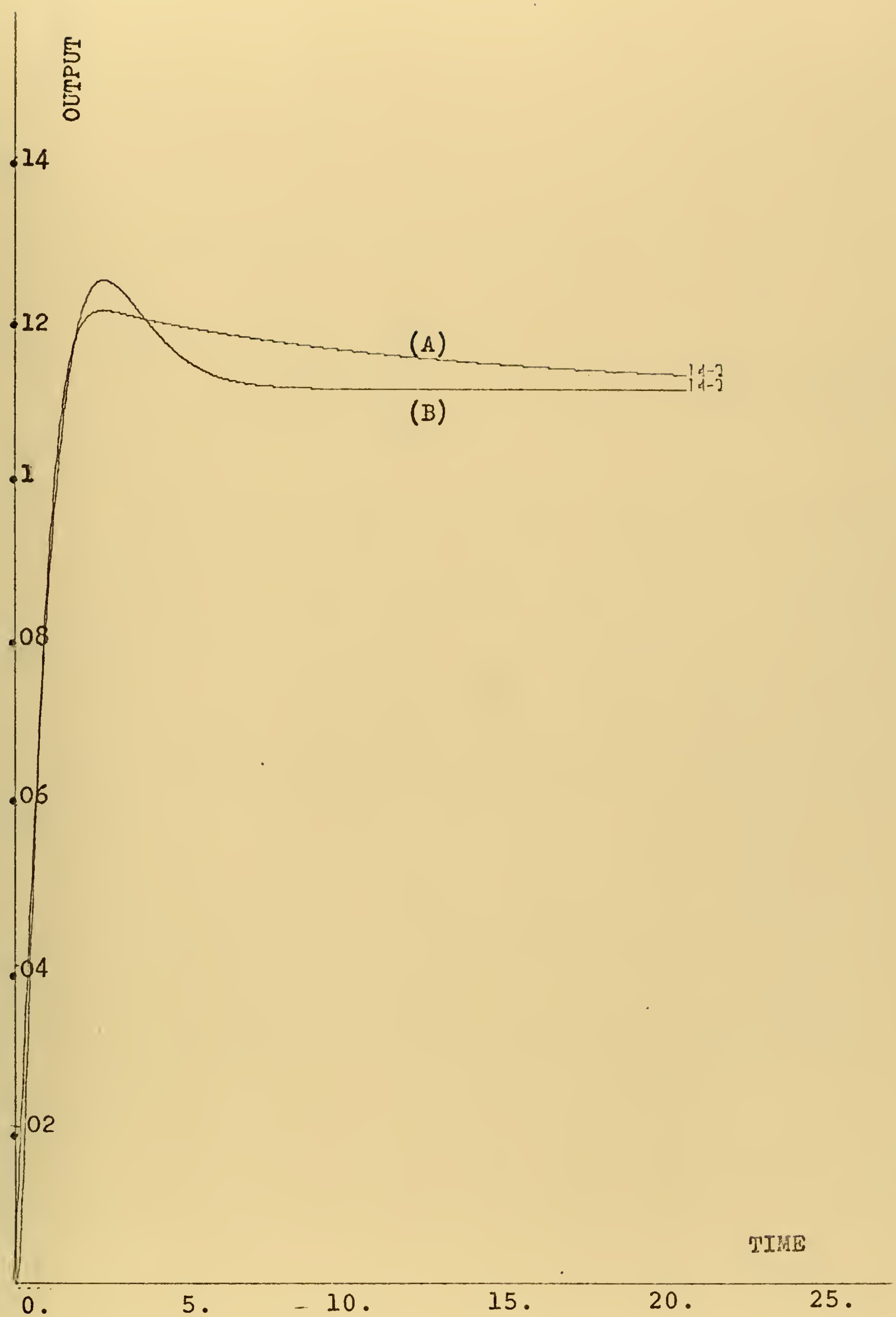

Figure III.57. EXAMPLE IV. System's response (A) and the Four Poles and Two Zeros model's response (B) to a unit step input.

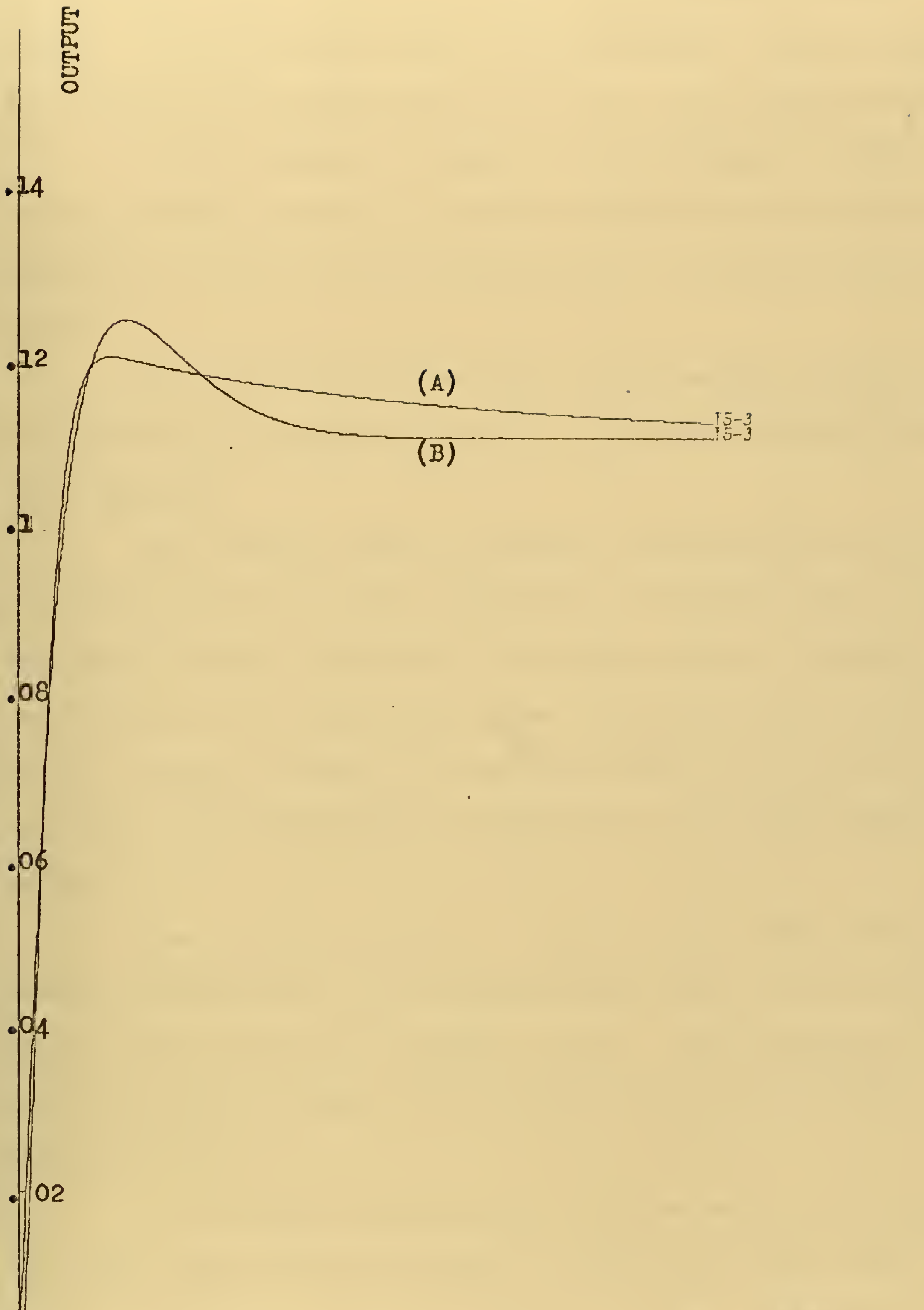

Figure III.58. EXAMPLE IV. System's response (A) and the Five Poles and Three Zeros model's response (B) to a unit step input.

to the origin appears to maintain its predominat effect mainly in the decaying part of the transient response and for that the determined models are not a good approximation.

It was decided to impose on the program a constraint on the possible values of the third real pole in order to maintain its value in the proximity of the original system's value.

The investigation of the reduced models for the control system of the pitch rate of a supersonic aircraft was then repeated.

The various models of second, third and fourth orders were obtained by the same technique as previous runs, using the same initial values for the complex poles. Besides constraining the third pole value to stay in the vicinity of its original value, in the models with one or more zeros a similar constraint was also imposed on the first zero value.

The optimum parameter values of the various new models obtained and the corresponding values of the performance index are shown in Table III.21. Step-responses of both original and lower order models are shown in Figs. III.59 to III.65.

It is worth noticing that the new models determined are much better approximations to the true system than the previous models. The cost function values are smaller and the third pole is close to the origin.

As a final checking of the necessity of constraining also the first zero in the minimization program in order to

Test System		.092	2.024 + j.965	2.024 - j.965	7.67 + j13.4	7.67 - j13.4	32. + j38.8	32. - j38.8	.083			
Poles	Zeros	p_1	p_2	p_3	p_4	p_5	p_6	p_7	z_1	z_2	z_3	$J \times 10^4$
2	0		1.17 + j1.4	1.17 - j1.4								4.7
	1		.54 + j.81	.54 - j.81					.293			3.6
3	0	.5	1.72 + j1.75	1.72 - j1.75								5.2
	1	.085	1.6 + j1.1	1.6 - j1.1					.078			.08
	2	.136	1.62 + j.79	1.62 - j.79					.119	33.1		.56
4	1	.091	3.1 + j.66	3.1 - j.66	3.66				.082			.03
	2	.101	2.45 + j1.2	2.45 - j1.2	2.194				.091	4.93		.07
	3	.077	1.42 + j.51	1.42 - j.51	1.73				.071	1.012	36.7	.2
7	1	.0894	1.84 + j1.2	1.84 - j1.2	6.44 + j13.5	6.44 - j13.5	39. + j36.9	39. - j36.9	.081			.01

TABLE III.21. EXAMPLE IV. New models.

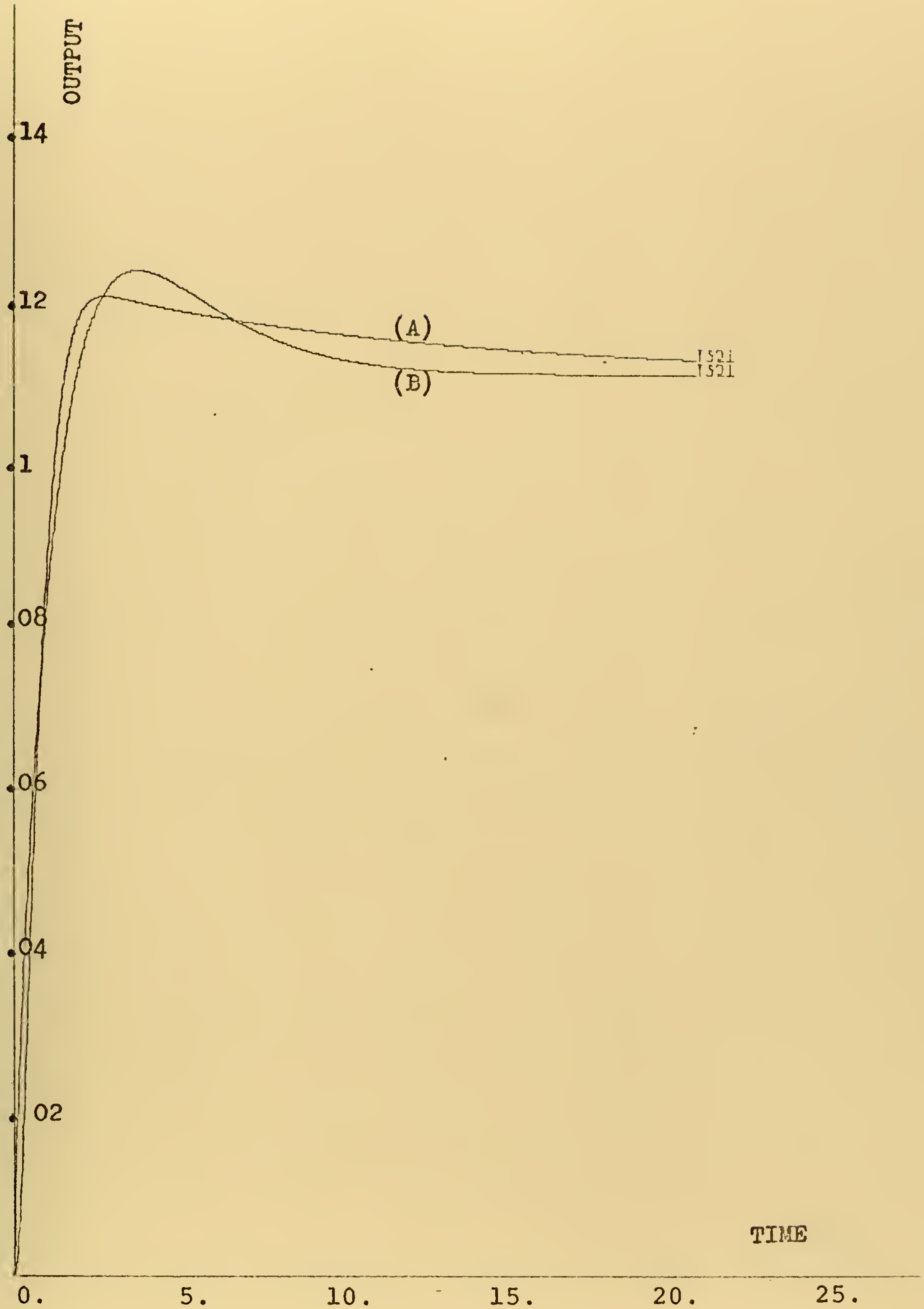

Figure III.59. EXAMPLE IV. System's response (A) and the Two Poles and One Zero model's response (B) to a unit step input.

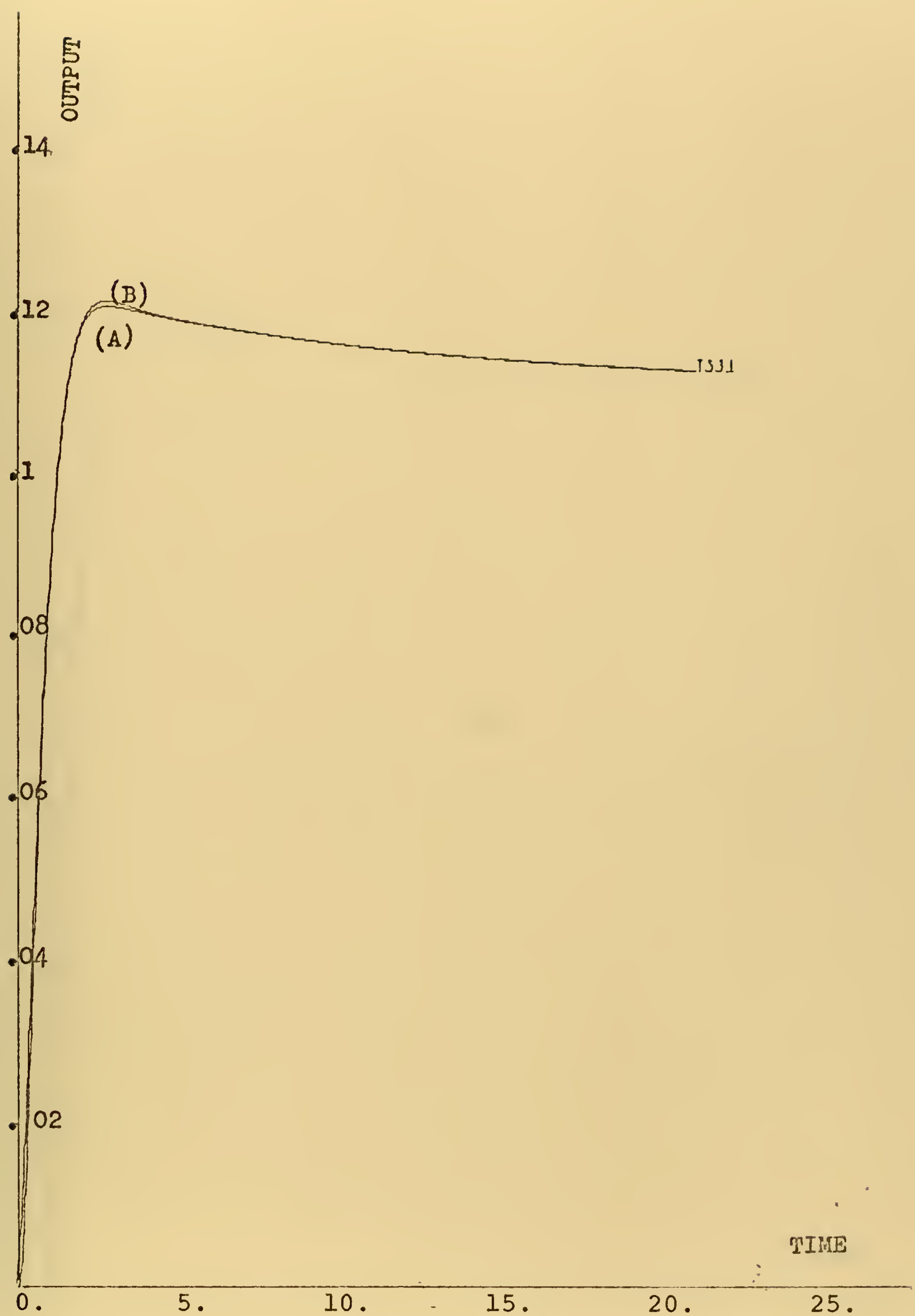

Figure III.60. EXAMPLE IV. System's response (A) and the Three Poles and One Zero model's response (B) to a unit step input.

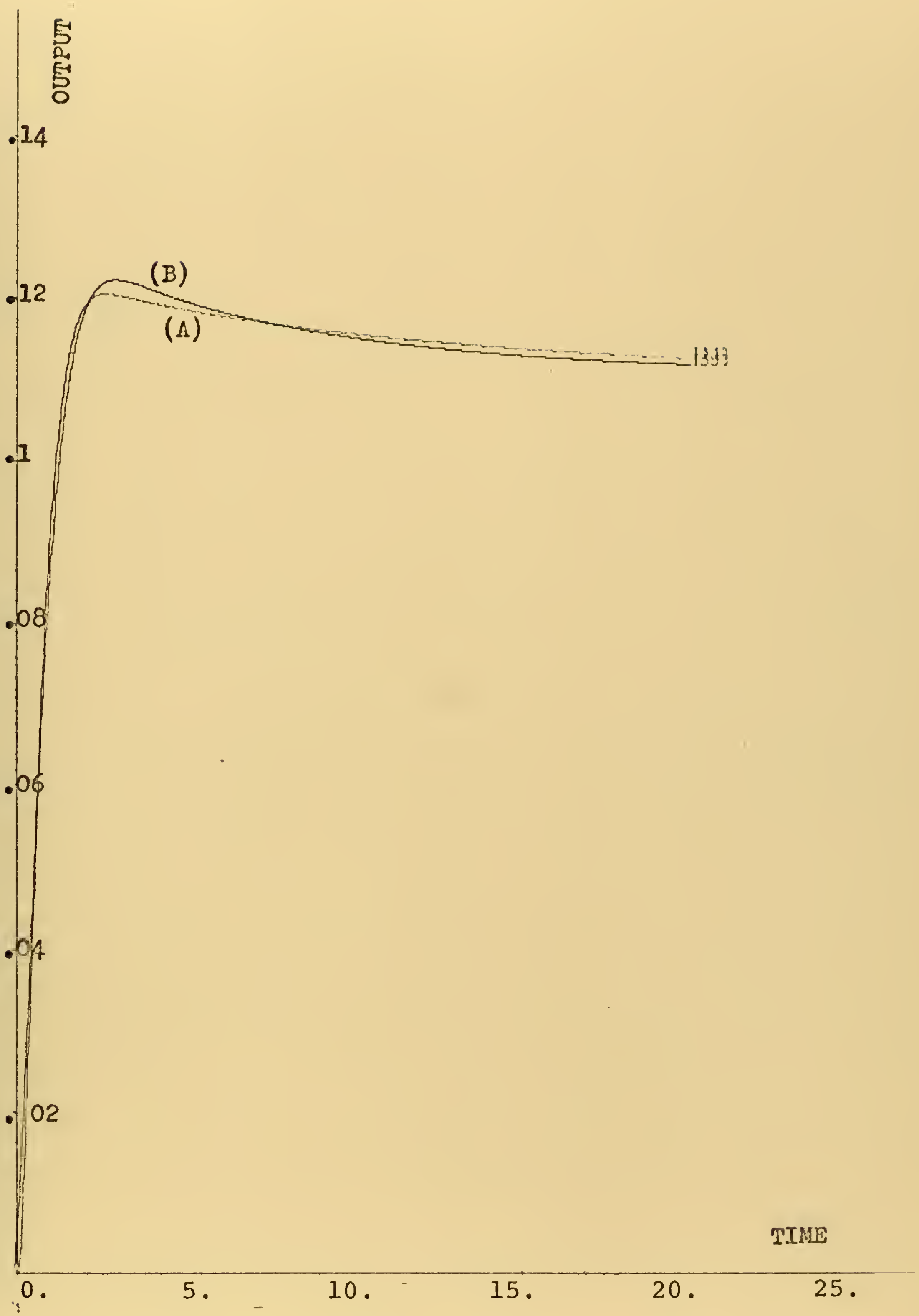

Figure III.61. EXAMPLE IV. System's response (A) and the Three Poles and Two Zeros model's response (B) to a unit step input.

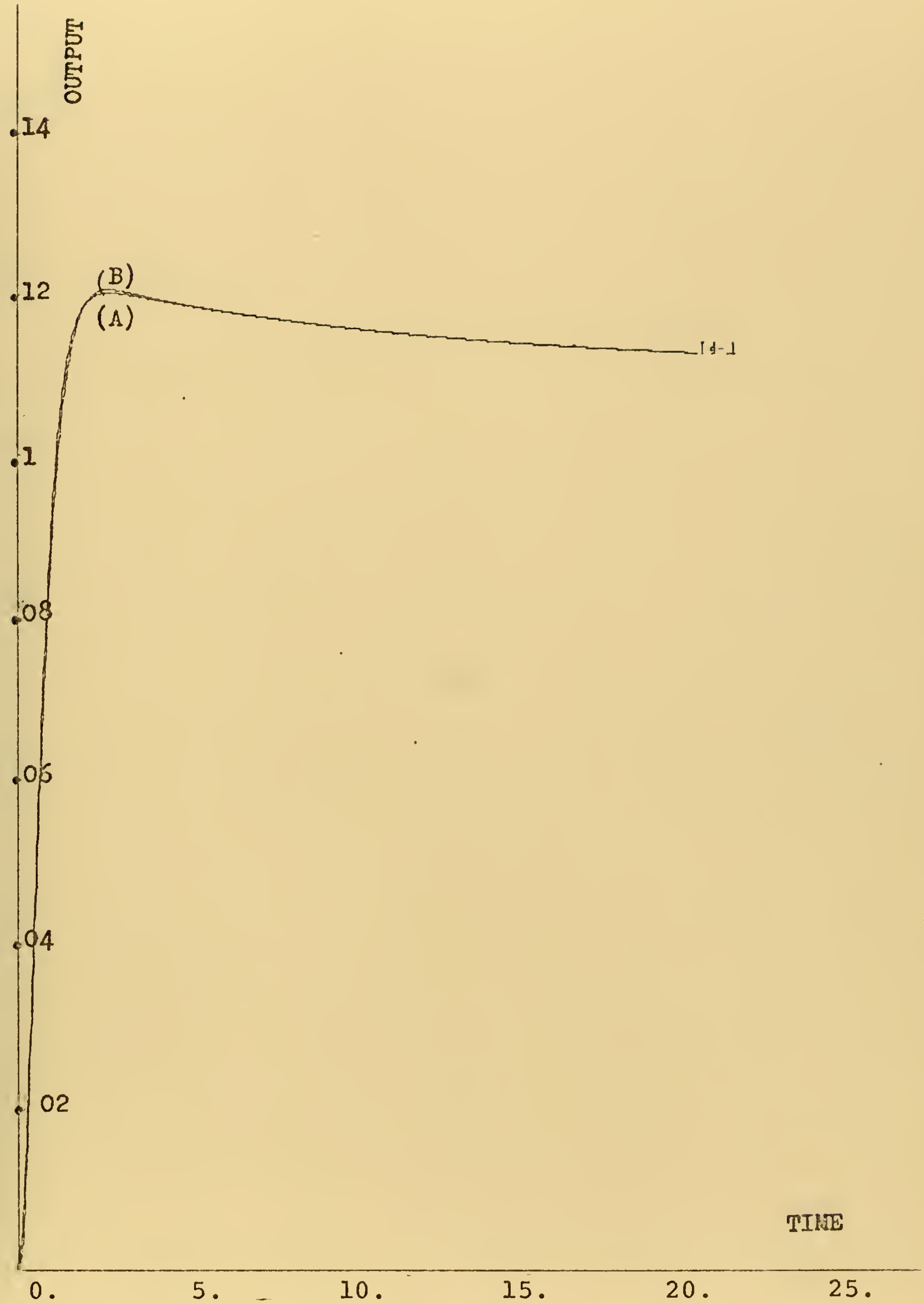

Figure III.62. EXAMPLE IV. System's response (A) and the Four Poles and One Zero model's response (B) to a unit step input.

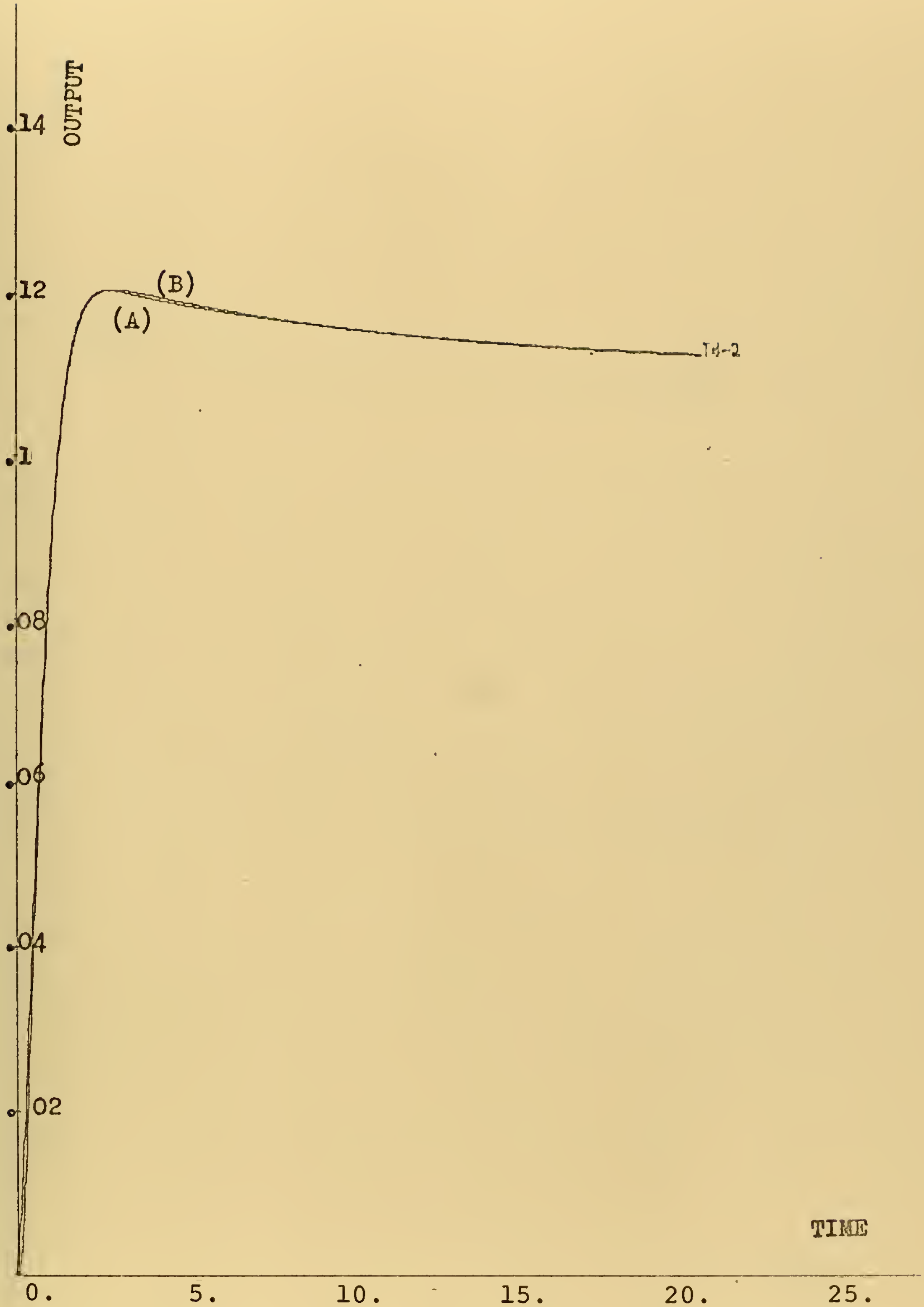

Figure III.63. EXAMPLE IV. System's response (A) and the Four Poles and Two Zeros model's response (B) to a unit step input.

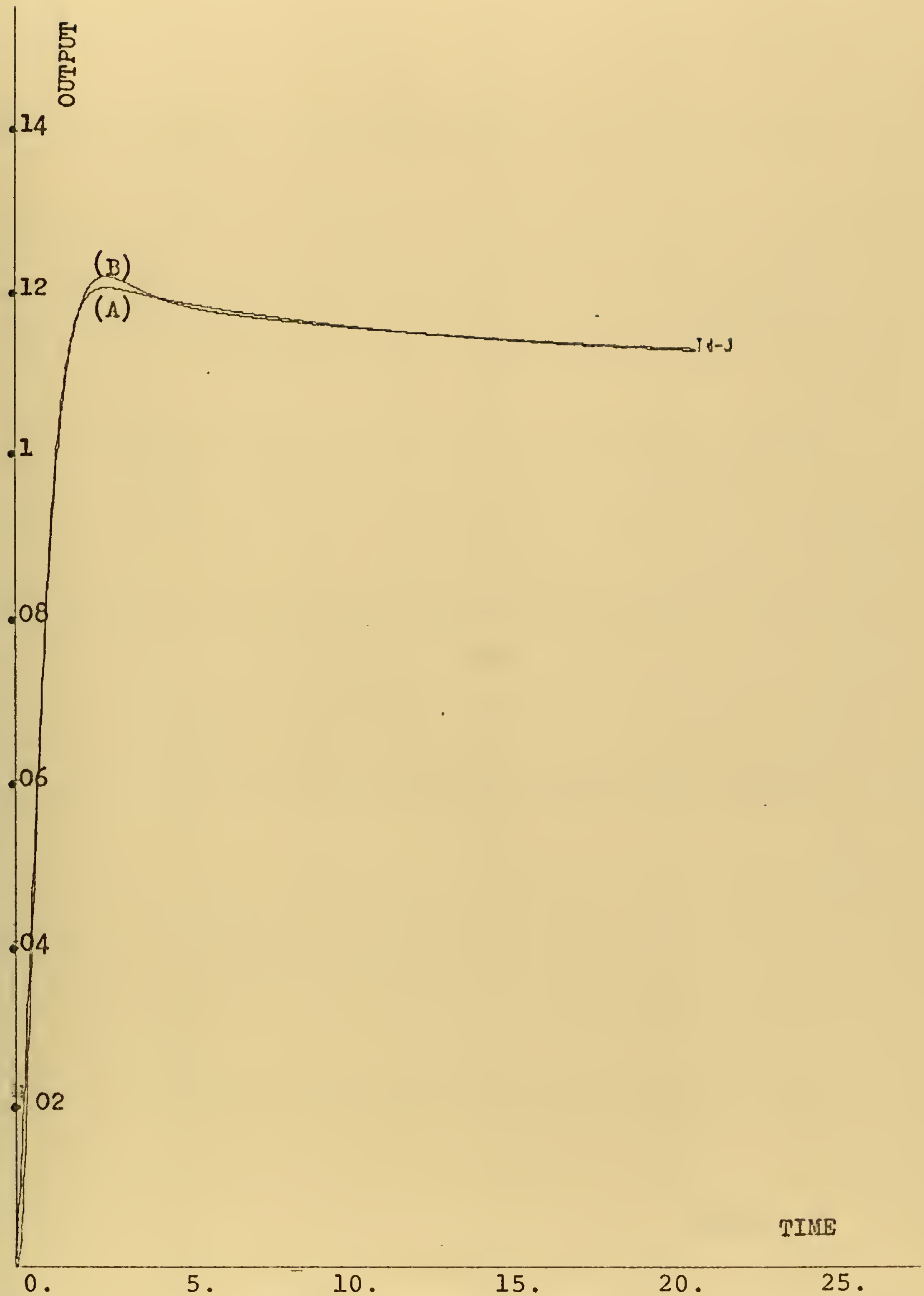

Figure III.64. EXAMPLE IV. System's response (A) and the Four Poles and Three Zeros model's response (B) to a unit step input.

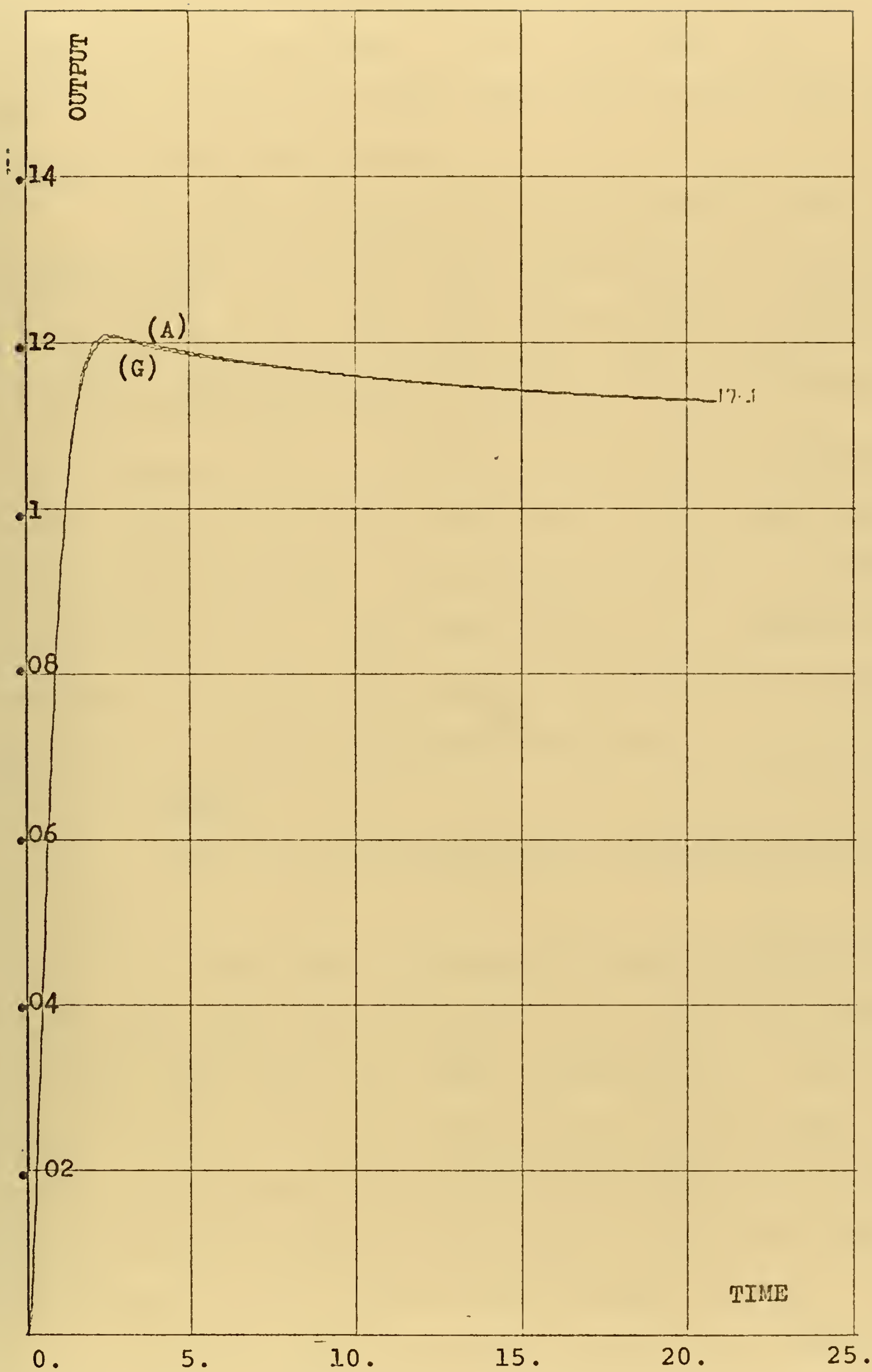

Figure III.65. EXAMPLE IV. System's response (A) and the Seven Poles and One Zero model's response (B) to a unit step input.

maintain its value in the proximity of the original system's value a new run was done. A new three poles and one zero model was determined setting constraint on the small pole but not on the small zero.

The optimum parameter values of the new model were:

New Model	p_1	p_2	p_3	z_1	$J\times10^4$
3 Poles 1 Zero	1.8+j0.84	1.8-j0.84	.094	.085	.09

which compared with the model obtained constraining both small pole and small zero (see Table III.21) does not appear to be much different. No constraint seems to be necessary on the small zero. Only seems mandatory to impose a constraint on the small pole upper bound because the decaying part of the system's transient response.

2. Remarks

Looking at the results the following comments should be noted:

a. The error criterion (J) value at Table III.20 are of the same order as the ones in Table III.5 (Example I) but the models are not a good approximation. The values of (J) at Table III.21 are much smaller and the models are a good approximation. It can be deduced that the cost function (J) is related mainly to the steady state final value which in Example I was 1.0 and in this example is 0.111. Selection of the desired model can be done taking into account this fact.

b. The second-order model still is worst and does not appear to be suitable.

c. The third and fourth order models with zeros achieve a very acceptable error criterion value.

d. A model which gives a minimum value of the performance index (J) is the four poles one zero model, although the three poles one zero and four poles two zeros give small value of (J). Again a "best" optimum model seems suitable.

e. The complex root values for the various models remain in a very small area. The first real pole and the first real zero remain in the proximity of the original ones. The system has one pole and one zero very close to the origin. It seems that in the cases in which the high-order system has singularities very close to the origin it is necessary to keep them in the low-order models.

f. Selection of the desired model can be done for specific required approximation depending on the desired simplicity, looking at the transient responses of both (original and approximated) and weighting the cost function values (J).

g. The starting second-order model assumed gives good starting parameter values for the derivation of increasing order models.

IV. CONCLUSIONS

The investigation presented can be considered as another approach to find the best low-order linear constant system which approximates the given high-order system.

It is based on the exhaustive use of the computer to estimate the free parameters which locate the roots of the model for which the integral of the squares of the errors is minimized. (The error is defined to be the difference between the response of the actual system and that of the model. The integral is evaluated between 0 and a selected final time.)

The description of the actual system dynamics need not necessarly be known. The case has been investigated in which the input-output data measured at discrete instants of time is known.

In Chapter III four examples were investigated to determine all the optimum models whose order varies from the second-order with no finite zeros to seventh-order with three real zeros. A complete set of graphs showing the variation of the error criteria value with respect to the number of zeros and poles of the model and all different pole and zero locations on the s-plane of the models were determined.

The variety of tables and figures has been obtained to enable the analyst and/or engineer to make a rapid and suitable choice of the order of the model weighting the

cost function (J). An analysis has been presented for evaluating the effects of such attempts in the various transient responses. It is felt that a new insight has been gained in the cost paid for the simplicity of the model and in the accuracy of the transient response of the model related to the numerical value of the error criterion, in order to get an acceptable response.

The following remarks can be stated:

1. The most common methods of modeling higher-order systems by making a second-order model approximation do not appear to be suitable for an error criterion value which give acceptable approximation according to the simplicity of the model.

With at least three poles and one zero models can be obtained which give a very good approximation.

2. A good approach for starting the study when the step-response of the system has an overshoot, was the determination of a pair of complex roots for the model derived from the curves of the step response of second-order linear system. To do this one takes into account the most common features of the step-response of the system.

When the transient response of the system shows the existence of a pole close to the origin, it appears to be mandatory to keep this singularity also in the lower order model.

3. It is possible, in effect, to weight the desired simplicity of the model versus selected error criterion

value. A "best" optimum low-order model which gives the minimum of all cost function values seems feasible.

4. The cost function value drops down very fast achieving a very acceptable value after the third or fourth order models are obtained. Then as the order of the model is increased the error criterion value varies very little and remains almost unchanged. So adding poles to the model - increasing its complexity or losing its simplicity - is not going to minimize the cost function value which remains in a very small area, i.e., the magnitude of the cost is essentially unchanged by the addition of poles and zeros, and furthermore keeps a very acceptable value.

Increasing the order of the model is not justified by the results because the price paid is high in comparison with the small improvement in the error criterion value. Under such circumstances the third or fourth order models seems to accomplish the characteristics of a good model. The decision in the choice of the reduced approximated model is done more easily knowing this fact in the cases in which limitations on the maximum order of the model or sufficiency of the resultant error (J) are real constraints.

5. The computer subroutine optimizes the selected model with respect to a specific error criterion. The user has to give the starting parameter values and it was seen how fundamental is this selection of values to achieve the optimum minimum error criterion. Limitations arise from this fact and from the convergence properties of the

algorithm which this subroutine uses. Improvement in efficiency should be possible by using a minimization program with known good convergence properties.

6. In Example IV was noted that in all cases the small pole and small zero did not go to the constraint limit, but they found a value within the constrained range. This seems to indicate that a local minimum does exist but was missed in the unconstrained searchs, probably because the minimization program used too large an increment.

As was stated in point 5, a good computer program will improve largely the efficiency of the search of the model.

```
C                 SAMPLE PROGRAM FOR EXAMPLE - I
C
C            OPTIMUM FOUR POLES AND ONE ZERO MODEL
C
C
      DIMENSION XS(5),BU(5),BL(5),XMN(5),XDATA(700)
      N=520
C   SAMPLED MEASURED INPUT-OUTPUT DATA OF THE SYSTEM
      READ(5,5) (XDATA(I),I=1,N)
    5 FORMAT(8E10.4)
      WRITE(6,6)
      WRITE(6,7) (XDATA(I),I=1,N)
      WRITE(6,8)
    6 FORMAT('1',T8,'THE INPUT TRANSIENT RESPONSE IS GIVEN BY :',////)
    7 FORMAT(2X,8E14.5)
    8 FORMAT(///T8,'SAMPLE INTERVAL IS : 0.01 SEC.',//////)
C   FREE PARAMETER VECTOR XMN(I)
C   UPPER BOUNDS
      BU(1)=0.8
      BU(2)=4.5
      BU(3)=20.
      BU(4)=80.
      BU(5)=100.
C   LOWER BOUNDS
      BL(1)=0.18
      BL(2)=1.5
      BL(3)=0.00001
      BL(4)=4.
      BL(5)=0.00001
C   STARTING POINT
      XS(1)=0.29
      XS(2)=3.15
      XS(3)=8.
      XS(4)=12.75
      XS(5)=22.
C   SUBROUTINE FOR MINIMIZATION
C   THE IMPORTANT OUTPUTS ARE:
C   XMN : VALUE OF THE FREE PARAMETERS AT A MINIMUM
C   YMN : VALUE OF THE PERFORMANCE INDEX
      CALL BOXPLX(5,0,50,0,0,XS,BU,BL,XMN,YMN,NMN,MMN,XDATA)
      WRITE(6,1) NMN,MMN
    1 FORMAT(//T8,2I6)
      WRITE(6,2) YMN
    2 FORMAT(//T8,'MINIMUM ERROR=',F10.5)
      WRITE(6,3)
    3 FORMAT(//T8,'OPTIMUM 4-TH MODEL')
      WRITE(6,4) XMN(1),XMN(2),XMN(3),XMN(4),XMN(5)
    4 FORMAT(//,10X,'ZETA=',F10.3,10X,'OMEGA=',F10.3,10X,'POLE1=',
```

```
     *F10.3,10X,'POLE2=',F10.3,10X,'ZERO=',F10.3,//)
      CALL SIMUL(XMN,XDATA)
    8 STOP
      END


      FUNCTION FE(A,XDATA)
C   CALCULATES THE VALUE OF THE PERFORMANCE INDEX
      DIMENSION A(5),XDATA(700)
      REAL*8 X(8),XDOT(5),T,DT
C   INITIAL CONDITIONS ARE ZERO
      DO 1 I=1,8
    1 X(I)=0.0
      T=0.0
      DT=0.01
      NT=0
      AA=2*A(1)*A(2)
      AB=A(3)+A(4)
      AC=A(3)*A(4)
      AD=A(2)**2
      ALFA=AA+AB
      BETA=AD+AC+AB*AA
      GAMMA=AA*AC+AB*AD
      DELTA=AC*AD
      EPSIL=DELTA/A(5)
      DO 7 I=1,520
C   SAMPLED INPUT-OUTPUT DATA OF THE SYSTEM
      X(8)=XDATA(I)
C   THE STATE EQUATIONS OF THE MODEL
    2 XDOT(1)=X(2)
      XDOT(2)=X(3)
      XDOT(3)=X(4)
      XDOT(4)=1.-DELTA*X(1)-GAMMA*X(2)-BETA*X(3)-ALFA*X(4)
      X(7)=DELTA*X(1)+EPSIL*X(2)
      X(6)=X(8)-X(7)
      XDOT(5)=X(6)**2
      S=RKLDEQ(5,X,XDOT,T,DT,NT)
      IF(S-1.) 4,2,7
    4 WRITE(6,5)
    5 FORMAT(//T8,'INTEGRATION TROUBLE')
      RETURN
    7 CONTINUE
C   VALUE OF THE PERFORMANCE INDEX
      FE=X(5)
      RETURN
      END
```

```
      FUNCTION KE(X)
C  SEE COMPUTATIONAL FLOW CHART FIG. (II.4)
C
      DIMENSION X(5)
      KE=0
      XX=X(1)*X(2)
      TS=5./XX
      IF(TS.GT.9.5) GO TO 12
      IF(TS.LT.3.) GO TO 12
      RETURN
   12 KE=-1
      RETURN
      END


      SUBROUTINE SIMUL(A,XDATA)
C  DIGITAL SIMULATION WITH OPTIMUM PARAMETERS OF
C  BOTH THE GIVEN SYSTEM AND THE REDUCED OPTIMUM MODEL
      DIMENSION XX(700),ZZ(700),YY(700),A(5),XDATA(700)
      REAL*8 X(4),XDOT(4),ITITLE(12)/'CANT 4-1',11*'        '/,T,DT
      REAL*4 LAB/'T4-1'/
      T=0.
      NT=0
C  SAMPLING AND INTEGRATION STEP SIZE
      DT=0.01
      DO 1 I=1,4
    1 X(I)=0.
      AA=2.*A(1)*A(2)
      AB=A(3)+A(4)
      AC=A(3)*A(4)
      AD=A(2)**2
      ALFA=AA+AB
      BETA=AD+AC+AB*AA
      GAMMA=AA*AC+AB*AD
      DELTA=AC*AD
      EPSIL=DELTA/A(5)
      DO 7 I=1,520
    2 XDOT(1)=X(2)
      XDOT(2)=X(3)
      XDOT(3)=X(4)
      XDOT(4)=1.-DELTA*X(1)-GAMMA*X(2)-BETA*X(3)-ALFA*X(4)
      S=RKLDEQ(4,X,XDOT,T,DT,NT)
      IF(S-1.) 5,2,6
    5 WRITE(6,4)
    4 FORMAT(//T8,'INTEGRATION TROUBLE')
      RETURN
    6 TT=I
      XX(I)=TT*DT
```

```
      YY(I)=XDATA(I)
      ZZ(I)=DELTA*X(1)+EPSIL*X(2)
    7 CONTINUE
C   PLOT OF THE SYSTEM'S TRANSIENT RESPONSE
      CALL DRAW(520,XX,YY,1,0,LAB,ITITLE,0,0,0,0,0,0,8,8,1,L)
      WRITE(6,8) L
    8 FORMAT(I6)
C   PLOT OF THE MODEL'S TRANSIENT RESPONSE
      CALL DRAW(520,XX,ZZ,3,0,LAB,ITITLE,0,0,0,0,0,0,8,8,1,L)
      WRITE(6,8) L
      RETURN
      END


      SUBROUTINE BOXPLX  (NV,NAV,NPR,NTZ,RZ,XS,BU,BL,XMN,YMN,NMN,MMN,
     *XDATA)
C
C     SUBROUTINE BOXPLX,FOR THE MINIMIZATION PROCESS,
C     IS LISTED IN THE SAMPLE PROGRAMFOR EXAMPLE - IV.
C
C
C     THE MEASURED INPUT-OUTPUT DATA OF THE HIGH-ORDER
C     SYSTEM,AT SAMPLING INTERVALS OF 0.01 SECONDS,ARE :
C
8.8541E-062.3632E-054.7185E-057.9231E-051.1173E-048.6123E-042.0568E-033.2102E-03
5.7420E-031.4200E-022.3301E-023.0200E-023.5983E-024.1767E-024.7548E-025.3332E-02
5.9116E-026.4900E-027.0684E-027.6468E-028.2251E-028.8035E-029.8395E-021.0875E-01
1.1911E-011.2947E-011.3983E-011.5019E-011.6055E-011.7091E-011.8200E-011.9163E-01
2.0200E-012.1615E-012.2645E-012.3498E-012.4405E-012.5320E-012.5915E-012.8874E-01
2.9831E-013.0794E-013.1898E-013.3012E-013.4130E-013.5664E-013.7398E-013.8525E-01
4.0518E-014.2531E-014.4561E-014.6605E-014.8662E-015.0729E-015.2804E-015.4885E-01
5.6968E-015.9054E-016.1138E-016.3220E-016.5297E-016.7367E-016.9428E-017.1479E-01
7.3517E-017.5541E-017.7550E-017.9541E-018.1513E-018.3463E-018.5392E-018.7297E-01
8.9177E-019.1030E-019.2856E-019.4652E-019.6418E-019.8153E-019.9855E-011.0152E 00
1.0316E 001.0476E 001.0632E 001.0784E 001.0933E 001.1078E 001.1219E 001.1356E 00
1.1489E 001.1617E 001.1742E 001.1862E 001.1979E 001.2090E 001.2198E 001.2301E 00
1.2400E 001.2495E 001.2585E 001.2671E 001.2753E 001.2830E 001.2903E 001.2972E 00
1.3036E 001.3096E 001.3152E 001.3204E 001.3254E 001.3294E 001.3333E 001.3368E 00
1.3399E 001.3426E 001.3449E 001.3469E 001.3484E 001.3496E 001.3504E 001.3508E 00
1.3509E 001.3506E 001.3500E 001.3490E 001.3478E 001.3462E 001.3443E 001.3421E 00
1.3396E 001.3368E 001.3337E 001.3304E 001.3268E 001.3229E 001.3188E 001.3145E 00
1.3099E 001.3052E 001.3002E 001.2950E 001.2897E 001.2841E 001.2784E 001.2725E 00
1.2665E 001.2603E 001.2540E 001.2475E 001.2410E 001.2343E 001.2276E 001.2207E 00
1.2138E 001.2067E 001.1997E 001.1925E 001.1853E 001.1781E 001.1708E 001.1636E 00
1.1563E 001.1489E 001.1416E 001.1343E 001.1270E 001.1197E 001.1125E 001.1052E 00
1.0981E 001.0909E 001.0838E 001.0768E 001.0698E 001.0629E 001.0561E 001.0494E 00
1.0427E 001.0361E 001.0297E 001.0233E 001.0170E 001.0108E 001.0048E 009.9884E-01
9.9302E-019.8732E-019.8175E-019.7630E-019.7099E-019.6581E-019.6077E-019.5586E-01
```

```
9.5110E-019.4647E-019.4200E-019.3767E-019.3348E-019.2945E-019.2556E-019.2183E-01
9.1824E-019.1481E-019.1153E-019.0840E-019.0543E-019.0261E-018.9994E-018.9742E-01
8.9505E-018.9284E-018.9077E-018.8886E-018.8709E-018.8547E-018.8399E-018.8266E-01
8.8147E-018.8042E-018.7951E-018.7874E-018.7810E-018.7760E-018.7723E-018.7698E-01
8.7687E-018.7687E-018.7700E-018.7725E-018.7761E-018.7809E-018.7868E-018.7938E-01
8.8018E-018.8109E-018.8210E-018.8320E-018.8440E-018.8569E-018.8706E-018.8852E-01
8.9006E-018.9168E-018.9338E-018.9514E-018.9698E-018.9888E-019.0084E-019.0287E-01
9.0494E-019.0708E-019.0926E-019.1149E-019.1376E-019.1607E-019.1842E-019.2080E-01
9.2322E-019.2566E-019.2813E-019.3062E-019.3313E-019.3566E-019.3820E-019.4075E-01
9.4331E-019.4587E-019.4844E-019.5100E-019.5357E-019.5613E-019.5868E-019.6123E-01
9.6376E-019.6627E-019.6877E-019.7126E-019.7372E-019.7616E-019.7857E-019.8096E-01
9.8332E-019.8565E-019.8794E-019.9021E-019.9243E-019.9462E-019.9678E-019.9889E-01
1.0010E 001.0030E 001.0050E 001.0069E 001.0088E 001.0107E 001.0125E 001.0142E 00
1.0159E 001.0176E 001.0192E 001.0208E 001.0223E 001.0237E 001.0251E 001.0265E 00
1.0278E 001.0290E 001.0302E 001.0313E 001.0324E 001.0335E 001.0344E 001.0353E 00
1.0362E 001.0370E 001.0378E 001.0385E 001.0392E 001.0398E 001.0403E 001.0408E 00
1.0413E 001.0417E 001.0420E 001.0423E 001.0426E 001.0428E 001.0430E 001.0431E 00
1.0432E 001.0432E 001.0432E 001.0431E 001.0430E 001.0429E 001.0427E 001.0425E 00
1.0422E 001.0419E 001.0416E 001.0413E 001.0409E 001.0404E 001.0400E 001.0395E 00
1.0390E 001.0384E 001.0378E 001.0372E 001.0366E 001.0360E 001.0353E 001.0346E 00
1.0339E 001.0331E 001.0324E 001.0316E 001.0308E 001.0300E 001.0292E 001.0284E 00
1.0275E 001.0267E 001.0258E 001.0250E 001.0241E 001.0232E 001.0223E 001.0214E 00
1.0205E 001.0196E 001.0187E 001.0178E 001.0169E 001.0160E 001.0151E 001.0142E 00
1.0133E 001.0125E 001.0116E 001.0107E 001.0098E 001.0090E 001.0081E 001.0073E 00
1.0064E 001.0056E 001.0048E 001.0040E 001.0032E 001.0024E 001.0017E 001.0009E 00
1.0002E 009.9945E-019.9874E-019.9804E-019.9737E-019.9671E-019.9606E-019.9543E-01
9.9482E-019.9423E-019.9365E-019.9309E-019.9255E-019.9202E-019.9152E-019.9103E-01
9.9056E-019.9012E-019.8969E-019.8927E-019.8888E-019.8850E-019.8815E-019.8781E-01
9.8749E-019.8719E-019.8691E-019.8665E-019.8641E-019.8618E-019.8597E-019.8578E-01
9.8561E-019.8546E-019.8532E-019.8520E-019.8510E-019.8501E-019.8494E-019.8489E-01
9.8485E-019.8483E-019.8483E-019.8484E-019.8486E-019.8490E-019.8495E-019.8502E-01
9.8510E-019.8519E-019.8530E-019.8541E-019.8555E-019.8569E-019.8584E-019.8601E-01
9.8618E-019.8637E-019.8656E-019.8677E-019.8698E-019.8720E-019.8743E-019.8767E-01
9.8792E-019.8817E-019.8843E-019.8870E-019.8897E-019.8925E-019.8953E-019.8982E-01
9.9011E-019.9040E-019.9070E-019.9101E-019.9131E-019.9162E-019.9193E-019.9224E-01
9.9256E-019.9287E-019.9319E-019.9350E-019.9382E-019.9413E-019.9445E-019.9476E-01
9.9508E-019.9539E-019.9570E-019.9601E-019.9632E-019.9662E-019.9692E-019.9722E-01
9.9752E-019.9781E-019.9810E-019.9838E-019.9866E-019.9894E-019.9921E-019.9948E-01
9.9974E-011.0000E 001.0002E 001.0005E 001.0007E 001.0010E 001.0012E 001.0014E 00
1.0016E 001.0019E 001.0021E 001.0023E 001.0025E 001.0027E 001.0028E 001.0030E 00
1.0032E 001.0033E 001.0035E 001.0036E 001.0038E 001.0039E 001.0041E 001.0042E 00
C     END
```

```
C                      SAMPLE PROGRAM FOR EXAMPLE -IV
C
C                 OPTIMUM FOUR POLES AND ONE ZERO MODEL
C
C
      DIMENSION XS(5),BU(5),BL(5),XMN(5),XDATA(700)
      REAL*8 X(7),XDOT(7),T,DT
C   FREE PARAMETER VECTOR XMN(I)
C   UPPER BOUNDS
      BU(1)=0.98
      BU(2)=3.5
      BU(3)=0.1
      BU(4)=5.
      BU(5)=0.4
C   LOWER BOUNDS
      BL(1)=0.5
      BL(2)=0.5
      BL(3)=0.0005
      BL(4)=0.1
      BL(5)=0.0001
C   STARTING POINT
      XS(1)=0.7
      XS(2)=2.1
      XS(3)=0.095
      XS(4)=0.8
      XS(5)=0.1
C   COEFFICIENTS OF THE CHARACTERISTIC EQUATION OF THE GIVEN SYSTEM
      C1=83.62
      C2=4097.
      C3=70342.
      C4=853704.
      C5=2814271.
      C6=3310875.
      C7=281200.
      C8=375000.
      C9=0.08333
      T=0.0
C   SAMPLING AND INTEGRATION STEP SIZE
      DT=0.03
      NT=0
C   INITIAL CONDITIONS ARE ZERO
      DO 9 I=1,7
    9 X(I)=0.0
      DO 7 I=1,700
C   THE STATE EQUATIONS OF THE SYSTEM
   12 XDOT(1)=X(2)
      XDOT(2)=X(3)
      XDOT(3)=X(4)
```

```
      XDOT(4)=X(5)
      XDOT(5)=X(6)
      XDOT(6)=X(7)
      XDOT(7)=1.-C7*X(1)-C6*X(2)-C5*X(3)-C4*X(4)-C3*X(5)-C2*X(6)-
     <C1*X(7)
      S=RKLDEQ(7,X,XDOT,T,DT,NT)
      IF(S-1.) 5,12,6
    5 WRITE(6,14)
   14 FORMAT(//T8,'INTEGRATION TROUBLE')
      GO TO 8
C   SAMPLED MEASURED INPUT-OUTPUT DATA OF THE SYSTEM
    6 XDATA(I)=C8*C9*X(1)+C8*X(2)
    7 CONTINUE
C   SUBROUTINE FOR MINIMIZATION
C   THE IMPORTANT OUTPUTS ARE:
C   XMN : VALUE OF THE FREE PARAMETERS AT A MINIMUM
C   YMN : VALUE OF THE PERFORMANCE INDEX
      CALL BOXPLX(5,0,50,0,0,XS,BU,BL,XMN,YMN,NMN,MMN,XDATA)
      WRITE(6,1) NMN,MMN
    1 FORMAT(//T8,2I6)
      WRITE(6,2) YMN
    2 FORMAT(//T8,'MINIMUM ERROR=',F10.5)
      WRITE(6,3)
    3 FORMAT(//T8,'OPTIMUM 4-TH MODEL')
      WRITE(6,4) XMN(1),XMN(2),XMN(3),XMN(4),XMN(5)
    4 FORMAT(//,10X,'ZETA=',F10.3,10X,'OMEGA=',F10.3,10X,'POLE1=',
     *F10.3,10X,'POLE2=',F10.3,10X,'ZERO=',F10.3,//)
      CALL SIMUL(XMN,XDATA)
    8 STOP
      END


      FUNCTION FE(A,XDATA)
C   CALCULATES THE VALUE OF THE PERFORMANCE INDEX
      DIMENSION A(5),XDATA(700)
      REAL*8 X(8),XDOT(5),T,DT
C   INITIAL CONDITIONS ARE ZERO
      DO 1 I=1,8
    1 X(I)=0.0
      T=0.0
      DT=0.03
      NT=0
      AA=2*A(1)*A(2)
      AB=A(3)+A(4)
      AC=A(3)*A(4)
      AD=A(2)**2
      ALFA=AA+AB
      BETA=AD+AC+AB*AA
```

```
      GAMMA=AA*AC+AB*AD
      DELTA=AC*AD
      EPSIL=DELTA/9.
      ETA=EPSIL/A(5)
      DO 7 I=1,700
C   SAMPLED INPUT-OUTPUT DATA OF THE SYSTEM
      X(8)=XDATA(I)
C   THE STATE EQUATIONS OF THE MODEL
    2 XDOT(1)=X(2)
      XDOT(2)=X(3)
      XDOT(3)=X(4)
      XDOT(4)=1.-DELTA*X(1)-GAMMA*X(2)-BETA*X(3)-ALFA*X(4)
      X(7)=EPSIL*X(1)+ETA*X(2)
      X(6)=X(8)-X(7)
      XDOT(5)=X(6)**2
      S=RKLDEQ(5,X,XDOT,T,DT,NT)
      IF(S-1.) 4,2,7
    4 WRITE(6,5)
    5 FORMAT(//T8,'INTEGRATION TROUBLE')
      RETURN
    7 CONTINUE
C   VALUE OF THE PERFORMANCE INDEX
      FE=X(5)
      RETURN
      END


      FUNCTION KE(X)
C   SEE COMPUTATIONAL FLOW CHART FIG. (II.4)
C
      DIMENSION X(5)
      KE=0
      RETURN
      END


      SUBROUTINE SIMUL(A,XDATA)
C   DIGITAL SIMULATION WITH OPTIMUM PARAMETERS OF
C   BOTH THE GIVEN SYSTEM AND THE REDUCED OPTIMUM MODEL
      DIMENSION XX(700),ZZ(700),YY(700),A(5),XDATA(700)
      REAL*8 X(4),XDOT(4),ITITLE(12)/'CANT 4-1',11*'        '/,T,DT
      REAL*4 LAB/'T4-1'/
      DO 1 I=1,4
      XDOT(I)=0.0
    1 X(I)=0.
      T=0.
      DT=0.03
      NT=0
```

```
      AA=2.*A(1)*A(2)
      AB=A(3)+A(4)
      AC=A(3)*A(4)
      AD=A(2)**2
      ALFA=AA+AB
      BETA=AD+AC+AB*AA
      GAMMA=AA*AC+AB*AD
      DELTA=AC*AD
      EPSIL=DELTA/9.
      ETA=EPSIL/A(5)
      DO 7 I=1,700
    2 XDOT(1)=X(2)
      XDOT(2)=X(3)
      XDOT(3)=X(4)
      XDOT(4)=1.-DELTA*X(1)-GAMMA*X(2)-BETA*X(3)-ALFA*X(4)
      S=RKLDEQ(4,X,XDOT,T,DT,NT)
      IF(S-1.) 5,2,6
    5 WRITE(6,4)
    4 FORMAT(//T8,'INTEGRATION TROUBLE')
      RETURN
    6 TT=I
      XX(I)=TT*DT
      YY(I)=XDATA(I)
      ZZ(I)=EPSIL*X(1)+ETA*X(2)
    7 CONTINUE
C   PLOT OF THE SYSTEM'S TRANSIENT RESPONSE
      CALL DRAW(700,XX,ZZ,1,0,LAB,ITITLE,0,0,0,0,0,0,8,8,0,L)
      WRITE(6,8) L
    8 FORMAT(I6)
C   PLOT OF THE MODEL'S TRANSIENT RESPONSE
      CALL DRAW(700,XX,YY,3,0,LAB,ITITLE,0,0,0,0,0,0,8,8,0,L)
      WRITE(6,8) L
      RETURN
      END

      SUBROUTINE BOXPLX  (NV,NAV,NPR,NTZ,RZ,XS,BU,BL,XMN,YMN,NMN,MMN,
     *XDATA)
      DIMENSION V(50,50), FUN(50), SUM(25), CEN(25), XS(25), BU(25),
     1    BL(25), XMN(25), U(2500),XDATA(700)
      EQUIVALENCE (V,U)
      KKON = 1                                                          BOXPL122
      IXR=50                                                            BOXPL123
      EP=1.E-7                                                          BOXPL124
      IF(NTZ) 1799,1799,1798                                            BOXPL125
 1799 NTA=2000                                                          BOXPL126
      GO TO 1797                                                        BOXPL127
 1798 NTA=NTZ                                                           BOXPL128
```

```
 1797 R=RZ
      IF(R) 1796,1796,1795                                                BOXPL129
 1795 IF(R-1.)1794,1796,1796                                              BOXPL130
 1796 R=1./3.                                                             BOXPL131
 1794 CONTINUE                                                            BOXPL132
      NVT=NV+NAV                                                          BOXPL133
      NT=0                                                                BOXPL135
      NPT=0                                                               BOXPL137
      NTFS=0                                                              BOXPL139
      DO 100 I=1,NV                                                       BOXPL143
      BL(I) = BL(I) + AMAX1 (EP, EP*ABS(BL(I)))                           BOXPL144
      BU(I) = BU(I) - AMAX1 (EP, EP*ABS(BU(I)))                           BOXPL145
      VT=XS(I)                                                            BOXPL146
      IF(BL(I)-VT)102,102,108                                             BOXPL147
  108 II = -I                                                             BOXPL148
      VT = BL(I)                                                          BOXPL149
      GO TO 101                                                           BOXPL150
  102 IF( BU(I) - VT) 110, 109, 109                                       BOXPL151
  110 II = I                                                              BOXPL152
      VT = BU(I)                                                          BOXPL153
  101 WRITE (6,104) II                                                    BOXPL154
  104 FORMAT(50HOINDEX AND DIRECTION OF OUTLYING VARIABLE AT START I5)    BOXPL155
  109 V(I,1) = VT                                                         BOXPL156
      CEN(I) = VT                                                         BOXPL157
      BL(I) = BL(I) + AMAX1(EP, EP * ABS ( BL(I)))                        BOXPL158
      BU(I) = BU(I) - AMAX1(EP   ,EP   *ABS (BU(I)))                      BOXPL159
  100 SUM(I)=VT                                                           BOXPL160
C                                                                         BOXPL161
      NCE=1                                                               BOXPL162
      I=1                                                                 BOXPL164
      IF (KE(V)) 162,106,162                                              BOXPL165
  162 IF (NPR) 161, 161, 163                                              BOXPL166
  163 WRITE (6,164)                                                       BOXPL167
  164 FORMAT (50HOIMPLICIT CONSTRAINT VIOLATED AT START. DEAD END.  )     BOXPL168
      GO TO 161                                                           BOXPL169
  106 NFE=1                                                               BOXPL170
      NLIM = 5*NV +100                                                    BOXPL172
      BESTFU =1.E70                                                       BOXPL174
      K=2*NV                                                              BOXPL175
      ALPHA=1.3                                                           BOXPL176
      FK=K                                                                BOXPL177
      FKM=FK-1.                                                           BOXPL178
      BETA=ALPHA+1.                                                       BOXPL179
      IQR = R * 1.E7                                                      BOXPL180
      IF (MOD(IQR,2).EQ.0) IQR=IQR+101                                    BOXPL181
      FUN(1) = FE(V,XDATA)
  103 FI=1.                                                               BOXPL185
      FUNOLD=FUN(1)                                                       BOXPL186
```

```
      DO 150 I=2,K                                                      BOXPL187
      FI=FI+1.                                                          BOXPL188
      DO 151 J=1,NV                                                     BOXPL189
      CALL RANDU (IQR,IQR,RQX)                                          BOXPL190
  151 V(J,I)=BL(J)+RQX      *(BU(J)-BL(J))                              BOXPL191
      DO 152 L=1,NLIM                                                   BOXPL193
      NCE=NCE+1                                                         BOXPL194
      IF (KE(V(1,I)))153, 157, 153                                      BOXPL195
  153 DO 154 J=1,NV                                                     BOXPL196
  154 V(J,I)=.5*(V(J,I)+CEN(J))                                         BOXPL197
  152 CONTINUE                                                          BOXPL198
      IF (NPR) 161,161,158                                              BOXPL199
  158 WRITE (6,155) I                                                   BOXPL200
  155 FORMAT(22HOCANNOT FIND FEASIBLE  I4,19HTH VERTEX AT START    )     BOXPL201
      CALL BOUT(NT ,NPT,NFE,NCE,NV,NVT,V,I,FUN,CEN )                    BOXPL202
  161 NMN = -I                                                          BOXPL203
      GO TO 1055                                                        BOXPL204
  157 DO 156 J=1,NV                                                     BOXPL205
      SUM(J)=SUM(J)+V(J,I)                                              BOXPL206
  156 CEN(J)=SUM(J)/FI                                                  BOXPL207
      NFE=NFE+1                                                         BOXPL208
      FUN(I) = FE(V(1,I),XDATA)
  150 CONTINUE                                                          BOXPL210
      IF (NPR) 159,159,160                                              BOXPL212
  160 CALL BOUT (NT,NPT,NFE,NCE,NV,NVT,V,K,FUN,CEN)                     BOXPL213
  159 FX = FNM (FUN, 0, K, MN)                                          BOXPL215
      M = MN                                                            BOXPL216
  208 FUNMAX = FNM(FUN, M, K, NM)                                       BOXPL220
      LIMT = 5*NV        +100                                           BOXPL223
      J=(M-1)*IXR                                                       BOXPL224
      J1=J+1                                                            BOXPL225
      DO 202 I=1,NV                                                     BOXPL226
      IJ=J+I                                                            BOXPL227
      VT=U(IJ)                                                          BOXPL228
      SUM(I) = SUM(I) - VT                                              BOXPL229
      CEN(I) = SUM(I)/FKM                                               BOXPL230
  202 U(IJ)=BETA*CEN(I)-ALPHA*VT                                        BOXPL231
      NT=NT+1                                                           BOXPL232
  213 IF(KKON) 2131,2131,2132                                           BOXPL234
 2132 DO 214 I=1,NV                                                     BOXPL235
      IJ=J+I                                                            BOXPL236
      VT=AMIN1(U(IJ),BU(I))                                             BOXPL237
  214 U(IJ)=AMAX1(VT,BL(I))                                             BOXPL238
 2131 DO 210 N=1,NLIM                                                   BOXPL239
      NCE=NCE+1                                                         BOXPL240
      IF(KE(U( J1)))207, 204, 207                                       BOXPL241
  207 DO 209 I=1,NV                                                     BOXPL242
      IJ=J+I                                                            BOXPL243
```

```
  209 U(IJ)=.5*(CEN(I)+U(IJ))                                                    BOXPL244
      NT=NT+1                                                                    BOXPL245
  210 CONTINUE                                                                   BOXPL246
 2100 IF(NPR) 2102,2102,2101                                                     BOXPL247
 2101 WRITE (6,221) NT, M                                                        BOXPL248
  221 FORMAT (10HOAT TRIAL  I4,29H CANNOT FIND FEASIBLE VERTEX  I4 ,             BOXPL249
     1  15X,  21HRESTART FROM CENTROID  )                                        BOXPL250
      CALL BOUT (NT,NPT,NFE,NCE,NV,NVT,V,K,FUN,CEN)                              BOXPL251
 2102 DO 2210 I = 1, NV                                                          BOXPL252
      SUM(I)=CEN(I)                                                              BOXPL253
 2210 V(I,1) = CEN(I)                                                            BOXPL254
      KQX=KE(V(1,1))
      NCE=NCE+1
      NFE=NFE+1                                                                  BOXPL255
      FUN(1) = FE(U(J1),XDATA)
      TFUN=FUN(1)                                                                BOXPL257
      IF (TFUN - BESTFU ) 2212, 2212, 2218                                       BOXPL258
 2218 IF(NPR) 1051, 1051, 2213                                                   BOXPL259
 2213 WRITE (6,2214)                                                             BOXPL260
 2214 FORMAT (27HOPREVIOUS MINIMUM WAS BEST.  )                                  BOXPL261
      GO TO 1051                                                                 BOXPL262
 2212 IF(NPR) 2215, 2215, 2216                                                   BOXPL263
 2216 WRITE (6,303) TFUN                                                         BOXPL264
 2215 YMN = TFUN                                                                 BOXPL265
      DO 2217 IL = 1, NVT                                                        BOXPL266
 2217 XMN(IL) = CEN(IL)                                                          BOXPL267
      ABFUN=ABS (TFUN-BESTFU )                                                   BOXPL268
      CRIFUN=AMAX1(ABS (TFUN)*EP,EP)                                             BOXPL269
      IF (ABFUN - CRIFUN) 1051,1051,2211                                         BOXPL270
 2211 BESTFU =TFUN                                                               BOXPL271
      GO TO 103                                                                  BOXPL272
  204 NFE=NFE+1                                                                  BOXPL274
      FUNTRY=FE(V(1,M),XDATA)
      IF(ABS (FUNTRY-FUNOLD)-AMAX1(ABS (EP   *FUNOLD),EP    ))217,217,           BOXPL277
     1  218                                                                      BOXPL278
  217 NTFS=NTFS+1                                                                BOXPL279
      IF(NTFS-K)211,300,300                                                      BOXPL280
  218 NTFS=0                                                                     BOXPL281
  211 IF(FUNTRY-FUNMAX)212,212,215                                               BOXPL282
  215 DO 216 I=1,NV                                                              BOXPL283
      IJ=J+I                                                                     BOXPL284
  216 U(IJ)=.5*(CEN(I)+U(IJ))                                                    BOXPL285
      LIMT=LIMT-1                                                                BOXPL286
      IF(LIMT) 2100,2160,2160                                                    BOXPL287
 2160 NT=NT+1                                                                    BOXPL288
      GO TO 213                                                                  BOXPL289
  212 FUN(M)=FUNTRY                                                              BOXPL291
      FUNOLD = FUNTRY                                                            BOXPL292
```

```
      NPT=NPT+1                                                          BOXPL293
      DO 203 I=1,NV                                                      BOXPL294
      IJ=J+I                                                             BOXPL295
  203 SUM(I)=SUM(I)+U(IJ)                                                BOXPL296
      M = NM                                                             BOXPL298
      IF (NPR) 400,400,401                                               BOXPL299
  401 IF( MOD (NPT,NPR)) 208,205,208                                     BOXPL300
  205 IYS = 1                                                            BOXPL301
  301 DO 222 IL = 1,NV                                                   BOXPL302
  222 CEN(IL) = SUM(IL)/FK                                               BOXPL303
      CALL BOUT(NT ,NPT,NFE,NCE,NV,NVT,V,K,FUN,CEN )                     BOXPL304
      GO TO (402,403), IYS                                               BOXPL305
  402 IF (NT - NTA) 208,219,219                                          BOXPL306
  219 WRITE (6,220)                                                      BOXPL307
  220 FORMAT (27HOLIMIT ON TRIALS EXCEEDED.  )                           BOXPL308
  403 TFUN = FE(CEN,XDATA)
      NFE=NFE+1                                                          BOXPL310
      WRITE (6,303) TFUN                                                 BOXPL311
  303 FORMAT(8HOMINIMUM,E20.7)                                           BOXPL312
      GO TO 1050                                                         BOXPL313
  400 IF (NT - NTA) 208,105,105                                          BOXPL314
  105 DO 1053 IL = 1,NV                                                  BOXPL315
 1053 CEN(IL) = SUM(IL)/FK                                               BOXPL316
      NFE=NFE+1                                                          BOXPL317
      TFUN = FE(CEN,XDATA)
 1050 YMN = TFUN                                                         BOXPL319
      DO 1052 I=1,NV                                                     BOXPL320
 1052 XMN(I)= CEN(I)                                                     BOXPL321
 1051 CALL KE(XMN)
      NMN = NT                                                           BOXPL325
      MMN = NPT                                                          BOXPL326
 1055 RETURN                                                             BOXPL327
  300 FK = FKM                                                           BOXPL328
      IF (NPR) 105,105,404                                               BOXPL329
  404 WRITE (6,302) K                                                    BOXPL330
  302 FORMAT (40HOFUNCTION HAS BEEN ALMOST UNCHANGED FOR I5, 7H TRIALS)  BOXPL331
      IYS = 2                                                            BOXPL332
      GO TO 301                                                          BOXPL333
      END                                                                BOXPL334

      FUNCTION FNM (FUN, M, K, NM)                                       BOXPL336
      DIMENSION FUN(50)                                                  BOXPL337
      FU = -1.E70                                                        BOXPL338
      DO 3 I = 1,K                                                       BOXPL339
      IF (I-M) 2, 3, 2                                                   BOXPL340
    2 IF (FU - FUN(I)) 1,3,3                                             BOXPL341
    1 FU = FUN(I)                                                        BOXPL342
```

```
      NM = I                                                            BOXPL343
    3 CONTINUE                                                          BOXPL344
      FNM = FU                                                          BOXPL345
      RETURN                                                            BOXPL346
      END                                                               BOXPL347

      SUBROUTINE RANDU (IX,IY,YFL)                                      BOXPL348
      IY=IX*65539                                                       BOXPL349
      IF(IY) 5,6,6                                                      BOXPL350
    5 IY = IY + 2147483647 + 1                                          BOXPL351
    6 YFL = IY                                                          BOXPL352
      YFL = YFL * .4656613E-9                                           BOXPL353
      RETURN                                                            BOXPL354
      END

      SUBROUTINE BOUT (NT,NPT,NFE,NCE,NV,NVT,V,K,FN,C)                  BOXPL357
      DIMENSION V(50,50),FN(50),C(25)                                   BOXPL358
      WRITE (6,1) NT,NPT,NFE,NCE                                        BOXPL359
    1 FORMAT (18HONUMBER OF TRIALS  I4,5X,20H PERMISSIBLE TRIALS  I4,   BOXPL360
     1 5X, 22H FUNCTION EVALUATIONS  I4,5X,24H CONSTRAINT EVALUATIONS   BOXPL361
     2 I4/ 1H0,25X, 21H- - - VERTICES - - -   )                         BOXPL362
      DO 4 I=1,K                                                        BOXPL364
      WRITE (6,2)  FN(I), (V(J,I), J=1,NV)                              BOXPL365
    2 FORMAT (1H , E18.7 ,2X,7E14.7 /(21X, 7E14.7))                     BOXPL366
      NVP=NV+1                                                          BOXPL367
    4 WRITE (6,3) (V(J,I),J=NVP,NVT)                                    BOXPL368
    3 FORMAT (21X,7E14.7)                                               BOXPL369
      WRITE (6,5) (C(I), I=1, NV)                                       BOXPL371
    5 FORMAT (10HOCENTROID  11X, 7E14.7/(21X,7E14.7))                   BOXPL372
      RETURN                                                            BOXPL373
      END                                                               BOXPL374
```

BIBLIOGRAPHY

1. Sinha, N. K. and Pillie, W., "A New Method for Reduction of Dynamic Systems," International Journal of Control, 1970.

2. Fellows, D. M., Sinha, N. K. and Wistmatch, J. C., Reduction of the order of Dynamic Systems," National Conference on Automatic Control, University of Waterloo, August 1970.

3. Sinha, N. K. and Bereznai, G. T., "Optimum Approximation of High-Order Systems by Low-Order Models," International Journal Control.

4. Kuppurajulu, A. and Elangovan, S., "System Analysis by Simplified Models," IEEE Transactions on Automatic Control, Vol. AC-15, April 1970.

5. Meier III, L. and Luenberger, D. G., "Approximation of Linear Constant Systems," IEEE Transactions on Automatic Control, Vol. AC-1, October 1967.

6. Meier, L., "Approximation of Linear Constant Systems With Linear Constant Systems of Lower Order," Ph.D. dissertation, Stanford University, Calif., May 1965.

7. Makovskii, V. A., "Determining Transfer Functions of Dynamic Units From Their Time Characteristics," Soviet Journal of Instrumentation and Control, Number 4, April 1967.

8. Davison, E. J., "A Method for Simplifying Linear Dynamic Systems," IEEE Transactions on Automatic Control, Vol. AC-11, 1966.

9. Chidambara, M. R. and Davison, E. J., "On a Method for Simplifying Linear Dynamic Systems," IEEE Transactions on Automatic Control, Vol. AC-12, 1966.

10. Mitra, D. "Analytic Results on the Use of Reduced Models in the Control of Linear Dynamic Systems," Proceedings IEE 116, 1969.

11. Anderson, J. H., "Geometrical Approach to the Reduction of Dynamic Systems," Proceedings IEE 114, 1967.

12. Patel, M. R. and Fallside, F., "Integral Performance Criteria," IEEE Transactions on Automatic Control, Vol. AC-10, October 1965.

13. Koppel, L. B., "Introduction to Control Theory With Applications to Process Control," Prentice-Hall, Inc., Englewood Cliffs, N. J.

14. Hilleary, R. R., "Subroutine Box PLX," NPG School Computer Facility Subroutine Library, January 1966.

INITIAL DISTRIBUTION LIST

		No. of Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Professor G. J. Thaler, Code 52Tr Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	5
4.	Exmo. Sr. Almirante Director de Enseñanza Naval Ministerio de Marina Madrid, Spain	1
5.	Ilmo. Sr. CN Director de la E.T.E.A. E.T.E.A. Vigo, Pontevedra, Spain	1
6.	Professor R. E. Bach, Jr., Code 52Bz Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
7.	Mr. D. R. Towill University of Wales Institute of Science and Technology Cathays Park, Cardiff. Wales	1
8.	Mr. Joseph Rubis NSRDC Annapolis, Maryland	1
9.	Mr. James Banham, Jr. Naval Ship Engineering Center Philadelphia Division, Sec 6700 Philadelphia, PA 19112	1
10.	Mr. James Garnett Naval Ship Engineering Center Attn: 6771 Philadelphia Division Philadelphia, PA 19112	1

11. Professor D. Javier Aracil 1
Escuela Tecnica Superior
de Ingenieros Industriales
Avenida Reina Mercedes, s/n
Sevilla, Spain

12. Professor D. J. Peracaula 1
Escuela Tecnica Superior
de Ingenieros Industriales
Departamento de Electronica
Avenida Generalisimo Franco, 999
Barcelona, Spain

13. TN. Don Angel Montero 1
E.T.E.A.
Vigo, Pontevedra. Spain

14. TN. Don Julio Cantalapiedra de la Gandara 4
Paseo Reina Cristina, 36. 3°
Madrid, 7. Spain

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* Naval Postgraduate School Monterey, California 93940	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

Low-Order Models for Dynamic Systems

4. DESCRIPTIVE NOTES *(Type of report and, inclusive dates)*

Master's Thesis; June 1972

5. AUTHOR(S) *(First name, middle initial, last name)*

Julio Cantalapiedra

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
June 1972	144	14

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
d.	

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Naval Postgraduate School Monterey, California 93940

13. ABSTRACT

An investigation directed at finding the best low-order model which approximates a given high-order system is presented. New insight is gained into the cost paid for the simplicity of the model and in the accuracy of the transient response of the model related to the magnitude of a cost function.

The problem is solved in the time domain by finding the best pole and zero locations of the model which minimize a defined error criterion. The computer is used to estimate these parameters, via a parameter minimizating program. A number of examples are included.

DD FORM 1 NOV 65 1473 (PAGE 1)

S/N 0101-807-6811

Security Classification

A-31408

KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Low-order Models						
Reduction of Dynamic Systems						
Linear Model Simplification						
Reduced Models						
Optimum Approximation						
Control						
Automatic Control						
Modeling						

135219
Thesis
C19443 Cantalapiedra
c.1 Low-order models for
dynamic systems.

19 FEB 74
18 APR 78
12 FEB 87
13 SEP 90
13 SEP 90

21492
25001
33307
[illegible]

Thesis 135219
C19443 Cantalapiedra
c.1 Low-order models for
dynamic systems.

thesC19443
Low-order models for dynamic systems.
3 2768 001 02009 2
DUDLEY KNOX LIBRARY